Notes on Transitioning from Excel 2003 Lists to Excel 2007 Tables





Paula Ecklund 2008-2009

Contents

| Introduction | |
|--|---|
| Background ······· 1 | |
| Why the change and Why these Notes2 | |
| The Excel 2003 List | |
| Characteristics of an Excel 2003 List3 | |
| What One Can Do with Excel 2003 List-Format Data5 | |
| The Excel 2003 Table and Array7 | |
| The Excel 2007 Table | |
| Overview 8 | |
| New Table Features/Organization in 2007 1 | |
| Automatic Autofilter1 | |
| Automatic Formatting1 | |
| Optional Total Row1 Dynamic Table Size1 | |
| Table Names (that are and are not really names)1 | |
| Internal Table Names1 | |
| Names for Table Elements & Writing Formulas | J |
| Using Internal Table Names1 | 4 |
| Excel Prompts for Internal Table Names | |
| Experiments Using Table Name Qualifiers1 | |
| Convert a Table Back to a List Range1 | 8 |
| Conversion Effects1 | 8 |
| Convert a Range (that was once a table) Back to a Table | 8 |
| Miscellaneous Table Notes | |
| When Names are Used/Not Used1 | |
| Odd Cell Behavior after Using a Table1 | 9 |
| My Conclusion1 | |
| The Excel 2007 Table-Like Range (2003 List)1 | |
| Use of the Term "Array" in 20072 | 0 |
| Use the New Excel 2007 Table Features vs. Stick with Excel | _ |
| 2007's 03-Like "List Range" 2 | U |

8/2008

Introduction

Background

Microsoft has stated that one of the main reasons for the Excel 2007 overhaul was to make more of Excel's useful features more easily accessible to the general user. Someone already familiar with pre-2007 versions of the software will not necessarily find that to be the case. Instead, it may feel more to them more like learning a strange variation of a once-familiar software application.

Excel users through Excel version 2003 accustomed themselves to Microsoft's specialized terms for data: *Database, Range, Array,* and *List*. In each case, one of these terms shared many, some, or *all* characteristics with one or more of the other terms. Still, each term had its own particular definition and was intended for use in particular Excel applications. Understanding Microsoft's intentions with regard to a term was important in order to use that term's features properly.

The terms have changed in Excel 2007. Here's quick summary of the old and the new.

| DATABASE | Structured data, or data arranged as records. The term <i>database</i> was used in Excel primarily before Microsoft added the Access relational database to the Office group. With Access as part of the Office group, Microsoft transferred the mantle of database features in Excel to the now specialized Excel term <i>list</i> . However the term <i>database</i> is still occasionally found in obscure places in post-Access Excel 2003. |
|----------|---|
| RANGE | A rectangular selection of cells or multiple rectangular selections of cells. Pre-2007 a range did not necessarily have anything to do with structured data. With 2007, structured data can be organized into an official "table" or exist as a structured data range (equivalent to the pre-2007 <i>list</i>). |
| ARRAY | Excel 2003 structured data used <i>without using the header row</i> (if a header row existed) could be referred to either as a <i>table</i> or as an <i>array</i> . (The term <i>array</i> had another, different meaning when its values were used as the object of calculations.) |

| LIST | The Excel 2003 name for structured data used in database operations such as sorts, LOOKUP functions, subtotals, Autoand Advanced Filters, and Pivot Tables. A 2003 <i>list</i> referred to data organized in records with each record holding information about a particular entity. A <i>list</i> was characterized by a first row of header labels, each column of the list contained the same kind of information, and there were no completely blank rows or completely blank columns in the list. |
|-------|---|
| TABLE | The Excel 2003 name for structured data used in database operations where a first row of header labels was <i>not</i> used. For example, INDEX operations used list-format data but operated on only the data part of a list, not using a header row. The data on which an INDEX function operated was distinguished from a <i>list</i> by the term <i>table</i> or <i>array</i> . Other than the use or non-use of the header row, a <i>list</i> and a <i>table</i> or <i>array</i> were organized in the same way. |
| | The 2007 name for a structured range of data that has been officially designated a <i>table</i> . A 2007 <i>table</i> automatically contains autofilter capability, is dynamic in size, and has a default name. A formula written in a table column cell can use "structured references" which can be automatically duplicated down the entire column. A final calculation row can be added to the <i>table</i> to summarize numeric columns. The term <i>list</i> no longer has meaning in Excel 2007. Excel 2003 operations used with <i>list</i> data are used with data organized as <i>tables</i> in Excel 2007. |

Why the Change? Why these Notes?

One of Microsoft's goals with Excel 2007 is to make Excel features more accessible to the average user. In the major Excel overhaul that resulted in Excel 2007, Microsoft changed the meaning and uses of the data terms used in Excel 2003 as well. A new Excel user starting out with the 2007 version will have no problem using the 2007 terms and features; they'll all be a part of learning a new software application. However, the user of Excel versions previous to 2007 will need to

- Learn where data management features are located in the new interface
- Carefully revise their understanding of Excel's changed terminology for data
- Understand how old Excel features are now grouped in Excel 2007
- Learn about the new features added to 2007 for working with Excel data
- Decide whether or not to use the new features or stick with the old ones
- Be aware of some of the traps and unexpected behaviors of features and techniques transformed from 2003 to 2007

The Excel 2003 List

Perhaps you're transitioning your Excel work environment from 2003 (or earlier) to 2007. Or, you may need to work in both environments for some period of time. This is a quick review of the Excel 2003 *list*.

Characteristics of the Excel 2003 List

The Excel 2003 *list* is a range of data that meets these specific criteria:

- The list range is rectangular (or square).
- The first row in the list contains column labels.
- Each row in the list is information about a particular entity or transaction. That is, each row/record provides information about one thing.
- Each column in the list holds the same kind of information.
- The list may have the occasional blank cell, but it has no completely blank rows or completely blank columns.

There are plenty of useful Excel data ranges that are not *lists*. *List* characteristics are important only when working with Excel features that expect structured data of this kind. Pivot Table operations, for example, can only work properly when the Pivot Table source data is organized as labeled records. The same is true for the Excel 2003 Autofilter and Advanced Filter and for many lookup operations.

A sample of data in Excel 2003 list format.

| | В | С | D | E | F | G | Н |
|----|--------|-------------------|----------|-------|------------------|----------|------------|
| 8 | REGION | MARKET | STORE | MODEL | LINE OF BUSINESS | REVENUE | Units Sold |
| 9 | North | Great Lakes | 65061011 | 4055T | Parts | \$3,454 | 6 |
| 10 | North | Shenandoah Valley | 62067017 | 2500P | Printer Sale | \$15,682 | 2 |
| 11 | North | Shenandoah Valley | 32139049 | 2500C | Copier Sale | \$12,968 | 6 |
| 12 | North | New England | 2004014 | 4055T | Parts | \$2,789 | 6 |
| 13 | North | New England | 72074014 | 4500C | Service Plan | \$32,605 | 23 |
| 14 | North | New England | 12011011 | 3002P | Service Plan | \$52,437 | 36 |
| 15 | North | New England | 2105015 | 2500P | Printer Sale | \$17,391 | 2 |
| 16 | North | New England | 22022012 | 4055T | Parts | \$2,468 | 6 |
| 17 | North | New England | 22022012 | 3002C | Parts | \$375 | 6 |

The same data but without a header row so not in Excel 2003 list format. This data can be used for Excel 2003 table/array operations where a header row is not used.

| | В | С | D | E | F | G | Н |
|----|-------|-------------------|----------|-------|--------------|----------|----|
| 9 | North | Great Lakes | 65061011 | 4055T | Parts | \$3,454 | 6 |
| 10 | North | Shenandoah Valley | 62067017 | 2500P | Printer Sale | \$15,682 | 2 |
| 11 | North | Shenandoah Valley | 32139049 | 2500C | Copier Sale | \$12,968 | 6 |
| 12 | North | New England | 2004014 | 4055T | Parts | \$2,789 | 6 |
| 13 | North | New England | 72074014 | 4500C | Service Plan | \$32,605 | 23 |
| 14 | North | New England | 12011011 | 3002P | Service Plan | \$52,437 | 36 |
| 15 | North | New England | 2105015 | 2500P | Printer Sale | \$17,391 | 2 |
| 16 | North | New England | 22022012 | 4055T | Parts | \$2,468 | 6 |
| 17 | North | New England | 22022012 | 3002C | Parts | \$375 | 6 |

The same data but not in Excel 2003 list format due to rogue data in the last column. Instead of "Units Sold", Rows 10:12 contain dates. This data could be used in list operations but might return inconsistent/incorrect results.

| | В | С | D | E | F | G | Н |
|----|--------|-------------------|----------|-------|------------------|----------|-----------------|
| 8 | REGION | MARKET | STORE | MODEL | LINE OF BUSINESS | REVENUE | Units Sold |
| 9 | North | Great Lakes | 65061011 | 4055T | Parts | \$3,454 | 6 |
| 10 | North | Shenandoah Valley | 62067017 | 2500P | Printer Sale | \$15,682 | May 15, 2008 |
| 11 | North | Shenandoah Valley | 32139049 | 2500C | Copier Sale | \$12,968 | June 29, 2008 |
| 12 | North | New England | 2004014 | 4055T | Parts | \$2,789 | August 12, 2008 |
| 13 | North | New England | 72074014 | 4500C | Service Plan | \$32,605 | 23 |
| 14 | North | New England | 12011011 | 3002P | Service Plan | \$52,437 | 36 |
| 15 | North | New England | 2105015 | 2500P | Printer Sale | \$17,391 | 2 |
| 16 | North | New England | 22022012 | 4055T | Parts | \$2,468 | 6 |
| 17 | North | New England | 22022012 | 3002C | Parts | \$375 | 6 |

The same data but not in Excel 2003 list format because all cells in Row 11 are blank. Any list operations performed on this data would operate only on the data in Rows 9 and 10.

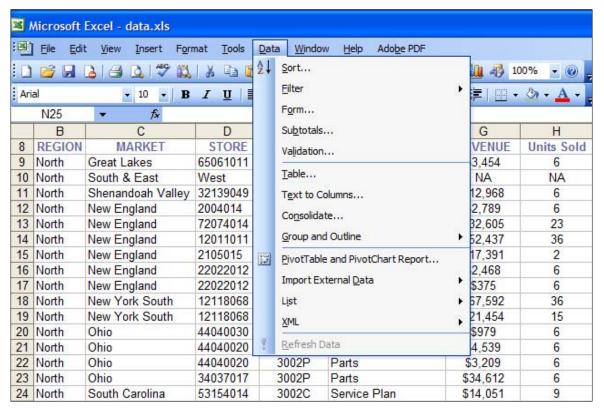
| | В | С | D | Е | F | G | Н |
|----|--------|-------------------|----------|-------|------------------|----------|------------|
| 8 | REGION | MARKET | STORE | MODEL | LINE OF BUSINESS | REVENUE | Units Sold |
| 9 | North | Great Lakes | 65061011 | 4055T | Parts | \$3,454 | 6 |
| 10 | North | Shenandoah Valley | 62067017 | 2500P | Printer Sale | \$15,682 | 2 |
| 11 | | | | | | | |
| 12 | North | New England | 2004014 | 4055T | Parts | \$2,789 | 6 |
| 13 | North | New England | 72074014 | 4500C | Service Plan | \$32,605 | 23 |
| 14 | North | New England | 12011011 | 3002P | Service Plan | \$52,437 | 36 |
| 15 | North | New England | 2105015 | 2500P | Printer Sale | \$17,391 | 2 |
| 16 | North | New England | 22022012 | 4055T | Parts | \$2,468 | 6 |
| 17 | North | New England | 22022012 | 3002C | Parts | \$375 | 6 |

The same data but not in Excel 2003 list format because Row 10 does not contain information about a particular entity or transaction. It appears to contain mixed and summary information. Again, this data could be used in list operations but would likely return inconsistent/incorrect results.

| | В | С | D | Е | F | G | Н |
|----|--------|-------------------|----------|-------|------------------|----------|------------|
| 8 | REGION | MARKET | STORE | MODEL | LINE OF BUSINESS | REVENUE | Units Sold |
| 9 | North | Great Lakes | 65061011 | 4055T | Parts | \$3,454 | 6 |
| 10 | North | South & East | West | All | All | NA | NA |
| 11 | North | Shenandoah Valley | 32139049 | 2500C | Copier Sale | \$12,968 | 6 |
| 12 | North | New England | 2004014 | 4055T | Parts | \$2,789 | 6 |
| 13 | North | New England | 72074014 | 4500C | Service Plan | \$32,605 | 23 |
| 14 | North | New England | 12011011 | 3002P | Service Plan | \$52,437 | 36 |
| 15 | North | New England | 2105015 | 2500P | Printer Sale | \$17,391 | 2 |
| 16 | North | New England | 22022012 | 4055T | Parts | \$2,468 | 6 |
| 17 | North | New England | 22022012 | 3002C | Parts | \$375 | 6 |

What One Can Do with Excel 2003 List-Format Data

In Excel 2003, most of the features for handling *list*-format data are located on the **Data** menu.



2003 Commands for Rearranging Data

Sort

Form (view data one record at a time)

Auto- and Advanced Filter

Text to Columns

Group and Outline

PivotTable and PivotChart Report

2003 Commands for Summarizing Data

Auto- and Advanced Filter

Subtotals

Consolidation

Group and Outline

PivotTable and PivotChart Report

2003 Commands for Working with External Data

PivotTable and PivotChart Report Import External Data

XML

Other 2003 Data Commands

<u>Validation</u> is in a class by itself. It provides a way to control and simplify data entry into a worksheet.

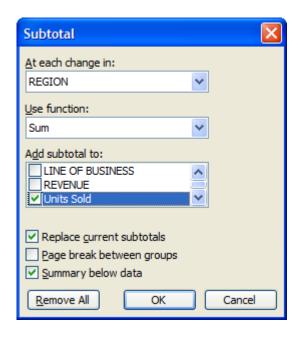
<u>Data Table</u> is an analysis/forecasting tool that works with a model and does not require data in list format.

<u>List</u> was Microsoft's first attempt to simplify and automate some of the *list* features. In 2003 its use was optional. The Excel 2007 *table* concept grew out of this early *list* option.

The <u>XML</u> option provides for specialized ways of working with Extended <u>Markup Language</u> for business-to-business communications.

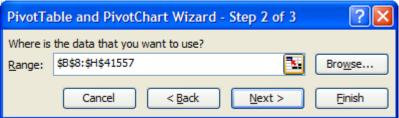
Excel 2003 data meeting the criteria for a *list* is immediately recognized as a *list* by some Excel operations. For example:

▶ If invoked when a single *list* cell is current, the Subtotal and the Sort automatically recognize the *list* range and "know" the name of each *list* column's label.





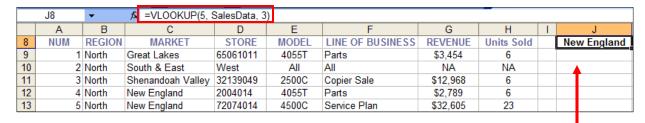
► If invoked when a single *list* cell is current, the PivotTable Wizard's Step 2 recognizes the extent of the list range automatically.



▶ If invoked when a single *list* cell is current, the data Form recognizes all the records in the list.



For operations that don't automatically recognize the Excel 2003 list, *naming the list range* helps makes its use straightforward. In the example below, the sample *list* has been range-named **SalesData**. The name SalesData is then used in a VLOOKUP function to retrieve data from one of the *list* record's cells.



=VLOOKUP(5, SalesData, 3)

Although the VLOOKUP function doesn't need the data headers, it understands that the first row in the *list* is header data not *list* data.

Excel's family of database functions (DSUM, DAVG, DMAX, etc.) is termed "dfunctions". They assume data is in list format.

The Excel 2003 Table and Array

Some data operations in Excel 2003 operate on structured data in records but do *not* understand column headers. For Excel 2003 operations like this, Microsoft refers to the data range not as a *list* but as a *table* or as an *array*.

In the example below the INDEX function retrieves data from the third row and the third column of the range named **noheaders**. The function expects the data range it operates on to be in structured format, just like a list, but with no header row.

| NUM | REGION | MARKET | STORE | MODEL | LINE OF BUSINESS | REVENUE | Units Sold | New England |
|-----|--------|-------------------|----------|-------|------------------|----------|------------|---|
| 1 | North | Great Lakes | 65061011 | 4055T | Parts | \$3,454 | 6 | |
| 2 | North | South & East | West | All | All | NA | NA | |
| 3 | North | Shenandoah Valley | 32139049 | 2500C | Copier Sale | \$12,968 | 6 | =index(noheaders, 3, 3) |
| 4 | North | New England | 2004014 | 4055T | Parts | \$2,789 | 6 | INDEX(array, row_num, [column_num]) |
| 5 | North | New England | 72074014 | 4500C | Service Plan | \$32,605 | 23 | <pre>INDEX(reference, row_num, [column_num], [area_num]</pre> |
| 6 | North | New England | 12011011 | 3002P | Service Plan | \$52,437 | 36 | / |
| 7 | North | New England | 2105015 | 2500P | Printer Sale | \$17,391 | 2 | |
| 8 | North | New England | 22022012 | 4055T | Parts | \$2,468 | 6 | |

The same sample list data used in the examples above is re-used here, but the data-only part of the range has been assigned its own name: **noheaders**

```
=index(noheaders, 3, 3)

INDEX(array, row_num, [column_num])
INDEX(reference, row_num, [column_num], [area_num])
```

Notice that the in the illustration above the tooltip prompt for the INDEX function uses the term **array** to prompt for the data-to-work-on. In Excel 2003 the terms *array* and *table* are often used interchangeably.

In Excel 2003 a range of cells that contain an array formula is also termed an array. This is a different meaning from the example above where the range contains structured data. Examples:

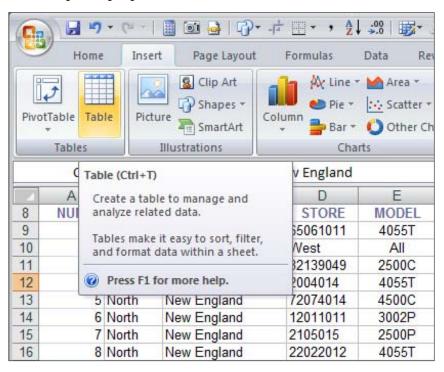
An array range: {3, 5, 9, 12, 15}

The content of a cell in a range that contains array formulas is: {=TABLE(,J12)}

The Excel 2007 Table

Overview

When using Excel 2007, it's important to understand the Excel 2007-specific term table. An Excel 2007 table is structured like an Excel 2003 list (header row, rows are records, cells in a column hold like data, no empty rows or columns). However, a big difference is that in Excel 2007 this list range can be deliberately designated a table and being a table confers special properties on the list.



Turning a list range into an official Excel 2007 table.

Excel 2007 has a "Data" tab on its ribbon. Some of the commands from the Excel 2003 Data menu option are located on this tab, but not all. Some table-related commands appear in more than one place. For example, "Sort & Filter" appears in the "Editing" group on the "Home" tab. In the "Sort & Filter" group of the "Data" tab are two separate commands: Sort and Filter.

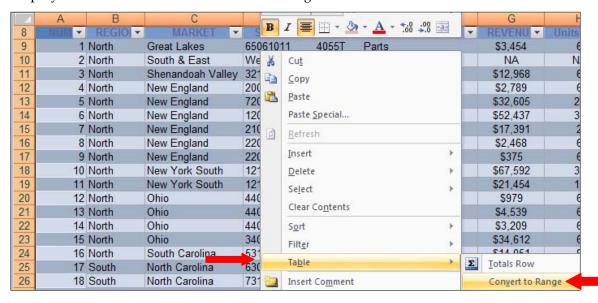
When you designate a list range an official 2007 *table*, a specialized "Table Tools"/"Design" tab appears when the *table* is active. The "Table Tools" tab has its own set of table-specific groups that hold table operations. The groups are **Properties**, **Tools**, **External Table Data**, **Table Style Options**, and **Table Styles**.



In another example of tool/icon duplication, the "Tools" group on the "Table Tools"/"Design" tab contains a command to start a Pivot Table. An insert Pivot Table option is also available on the "Insert" tab in the "Tables" group.

Note: How to turn an Excel 2007 *Table* back into a regular range

Before going further, here's how to turn a designated Excel 2007 *table* back to a regular Excel 2007 list range: Right-click any cell in the table. From the context menu that displays choose *Table* and then *Convert to Range*.



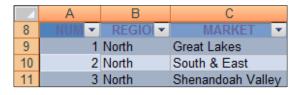
New Table Features/Organization in Excel 2007

If - like me -you declined to use the Excel 2003 "list precursor" to the Excel 2007 *table*, then the features of the Excel 2007 *table* described below are either new and/or organized in a new way.

Automatic Autofilter

As soon as you designate a range an Excel 2007 *table*, the Autofilter automatically turns on.

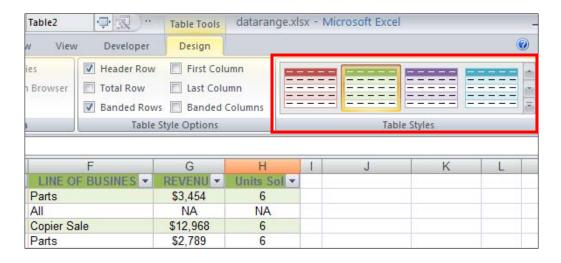
Notice, however, that you can turn the Autofilter on manually to use with a range *not* officially designated as an Excel 2007 *table*.



| | Α | В | С | |
|----|------|---------|-------------------|--|
| 7 | | | | |
| 8 | NUM▼ | REGIO ▼ | MARKET 💌 | |
| 9 | 1 | North | Great Lakes | |
| 10 | 2 | North | South & East | |
| 11 | 3 | North | Shenandoah Valley | |

Automatic Formatting

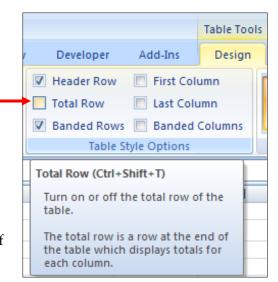
An Excel 2007 *table* automatically has default formatting assigned. Change the format using the "Table Styles" group on the "Table Tools Design" tab. The upper-left-most format option named "Light" basically displays your *table* with no formatting. In this case (especially if the Autofilter is turned off) your *table* might be mistaken for a simple range. However, unless you've deliberately converted the *table* to a regular range, the Excel 2007 *table* features remain in effect.



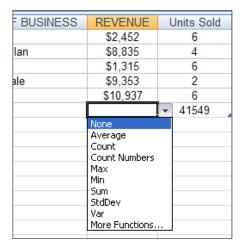


In the "Table Style Options" group of the "Table Tools Design" tab is a "Total Row" option. Checking this option on turns on a total row as the last row of your table and automatically goes to that row.

The total row default is the label "Total" in the left-most column of the row and a sum of the right-most column of the table.



| | NUM | REGION | MARKET | STORE | MODEL | LINE OF BUSINESS | REVENUE | Units Sold |
|-------|---------|--------|------------|----------|-------|------------------|----------|------------|
| 41553 | 41545 | West | North West | 57054024 | 3002P | Parts | \$2,452 | 6 |
| 41554 | 41546 | West | North West | 81183023 | 5001P | Service Plan | \$8,835 | 4 |
| 41555 | 41547 | West | North West | 81183013 | 3002C | Parts | \$1,315 | 6 |
| 41556 | 41548 | West | Central | 66063013 | 2500P | Printer Sale | \$9,353 | 2 |
| 41557 | 41549 | West | Central | 96193023 | 3002P | Parts | \$10,937 | 6 |
| 41558 | Total 🛑 | | | | | | | 41549 |
| 41559 | | | | | | | | |



Unless you click a cell in the total row, it's not obvious what other options are available. In the illustration at left, clicking in the total row under the "Revenue" column displays a drop-down list of calculations that can be applied to the table data in

41553

41554

41555

MUM

Average

Count

Min

Sum

41565 Var

41567

StdDev

41566 More Functions.

41545 West

41546 West

41547 West

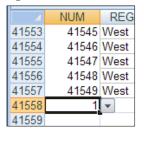
41548 West

41549 West

that column.

41556 41557 41558 Total 41559 None 41560 41561 Count Numbers 41562 Max 41563 41564

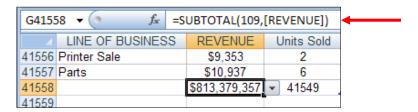
Despite the fact that the first cell in the total row continues to display the label "Total", that cell also has a drop-down of options.



At left I asked the total row to calculate the MIN for the data in the left-most column.

REGION

If you're familiar with the way Excel 2003 handled calculations on filtered *list* data, you'll understand the functions Excel 2007 uses to generate values on its *table* total row. Even if you turn off the Autofilter for the Excel 2007 table, any calculation you ask Excel to generate on the total row uses the SUBTOTAL function. (This function is different from the "Subtotal" option in the "Outline" group on the "Data" menu.) In brief, Excel uses the SUBTOTAL function and its multiple parameters to achieve calculations that would normally be generated by functions such as SUM, AVG, MAX, and MIN.



The SUBTOTAL function operates on visible cells only. *If* the *table* Autofilter has been used to hide one or more rows of data, you can be sure that the total row calculations return values for the visible rows only. Again, in Excel 2007, the SUBTOTAL function is used whether or not the *table* data has been autofiltered. Since Excel 2007 generates summary *table* calculations for you on demand, understanding the function it uses is primarily a matter of clarification.

Using the *table* total row to attempt to apply a numeric summary operation to a column containing text returns a regular Excel error message. However, COUNT can be applied to such a column. In the example at right, Excel uses this formula:

| MODEL | LINE OF BUSINESS | F | REVENUE | |
|-------|----------------------|---------|------------|--|
| 2500P | Printer Sale | \$9,353 | | |
| 3002P | Parts | | \$10,937 | |
| | 41549 | v | 13,379,357 | |
| | None Average | | | |
| | Count | | | |
| | Count Numbers Max | | | |

=SUBTOTAL(103,[LINE OF BUSINESS])

Where the parameter 103 indicates that a COUNT operation should be performed.

In Excel 2003, it was incumbent on the Excel <u>user</u> to be aware of the potential dangers of applying regular functions to filtered data and to use the SUBTOTAL function if necessary. Excel 2007 has removed that potential danger source by imposing the rule that a 2007 *table* total row will <u>always</u> calculate using <u>only</u> visible cells.

Dynamic Table Size

Entering data in <u>any</u> cell in a column to the right of the right-most existing Excel 2007 *table* column automatically expands the *table* one column wider. A default column header name is assigned unless you enter your own.

When positioned in the last cell of the last row of data, hitting the TAB key automatically inserts a new row into the *table*. If the *table* has the total row turned on, the new row is inserted above the total row.

One or more rows and columns can be inserted <u>within</u> an existing *table* in the usual way.

Using any of these methods to resize an Excel 2007 *table* (whether to expand <u>or</u> contract) does not require the Excel user to redefine the range the *table* occupies on the worksheet. Excel 2007 keeps track of the *table*'s dimensions automatically. By contrast, unless a user named an Excel 2003 table range <u>dynamically</u> (using a formula employing the OFFSET and COUNT functions) any records added to the <u>end</u> of an Excel 2003 *list* required a redefinition of the *list* range. And, even a dynamic name for a *list* did not exempt the 2003 user from redefining the *list* range in the case of a column addition to the right of the existing *list*'s right-most column.

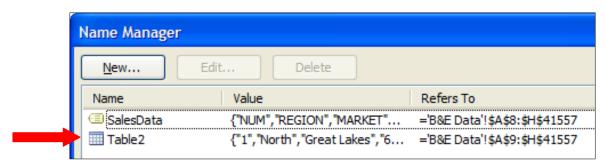
Table Names (that are and are not really names)

One of the most interesting/confusing features of the Excel 2007 *table* is the automatic names Excel generates for the *table* itself and for columns of the *table*. As an Excel user, you can also generate your own names for the *table* range and for the *table* columns and/or refer to the *table* or columns of the *table* using the regular Excel Row/Column nomenclature (such as A1:F200 for an entire range or A1:A200 for a first column).

The best way to see what the operation of Excel 2007 automatic *table* names and the names you assign as you use the *table* is to experiment. Descriptions of some of my own experiments follow. You may not necessarily find Excel's treatment of names intuitive, especially if you convert the *table* back to a regular range and vice-versa.

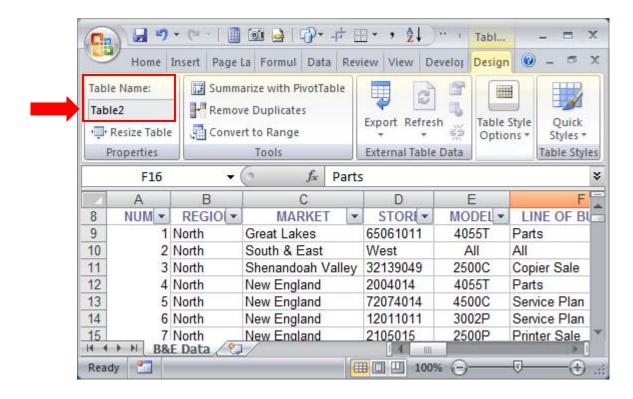
Internal *Table* Names

When you designate a list range as an Excel 2007 *table* Excel automatically assigns the table a name. Excel names *tables* in the same way it names worksheets. That is, the first *table* created in a workbook is named **Table1**, the next is named **Table2**, and so on. Even if **Table1** is deleted and a new *table* is created, Excel will not re-use the name **Table1** in the workbook. The new *table* becomes **Table3**.



As illustrated above, a *table* can have a user-created name as well as the Excel default name. Above, the same range of data is named both **Table2** and **SalesData**.

Opening the "Name Manager" dialog from the "Defined Names" group on the "Formulas" tab is a good way to view and manipulate range names in an Excel 2007 workbook. In the "Properties" group of the "Table Tools Design" tab, the name of the current *table* also is shown.



Names for *Table* Elements & Writing Formulas Using Internal *Table* Names If you create a formula in the worksheet that references *table* cells <u>and you create</u> the formula by typing and using the arrow keys to select a table cell or cells, then Excel automatically generates the formula using the default *table* name along with special name-like references to table elements. In the illustration below, the formula in Cell J9 was created by typing an equals sign, arrowing to the left and clicking Cell H9, typing *1.12 and hitting the enter key.

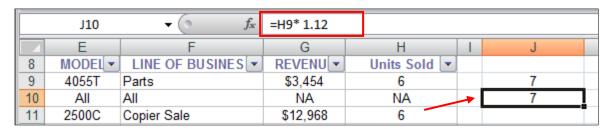
| | J9 | ▼ (f _x | =Table2[[#Th | nis Row],[Units Sol | d]]* | 1.12 |
|----|---------|---------------------------|-----------------|---------------------|------|------|
| | Е | F | G | Н | 1 | J |
| 8 | MODEL - | LINE OF BUSINES | REVENU ▼ | Units Sold 💌 | | |
| 9 | 4055T | Parts | \$3,454 | 6 | | 7 |
| 10 | All | All | NA | NA | | |
| 11 | 2500C | Copier Sale | \$12,968 | 6 | | |

The name **Table2** identifies which table. Excel encloses in square brackets any references to elements inside the table. Hence **[Units Sold]**, referring to Cell H9. Excel has added the clarifier **[#This Row]**, in front of the cell reference. The *table* name parameters are enclosed in outer square brackets.

Note

Although you (and Excel) can refer to any *table* element (such as a table column) by using its header label in square brackets, none of the *table* element names appear in Excel's "Name Manager" dialog. They can be <u>used</u> as names in calculations but they don't exist as <u>real</u> names. More below.

You can get the same answer *without* using the internal names by writing the formula in a more conventional way. That is, typing the equals sign, typing **H9*1.12** (or clicking Cell H9 and typing *** 1.12**) and hitting the enter key. In this case, Excel does <u>not</u> use default *table* names.



If you're familiar with Excel 2007's *table* names scheme, you can type a formula yourself and use the default names. For example, you might type:

=Table2[[#This Row],[Units Sold]]*1.12

<u>Surprisingly</u>, this formula does not adjust what it references if its location is <u>moved</u>. The designator/qualifier [#This Row] evidently continues to refer to the row it originally referenced at the time the formula was created. (This behavior is different from the way a normal relative formula cell reference works when a formula is moved in a worksheet.) As ambiguous as it is, the qualifier [#This Row] acts like an absolute reference.

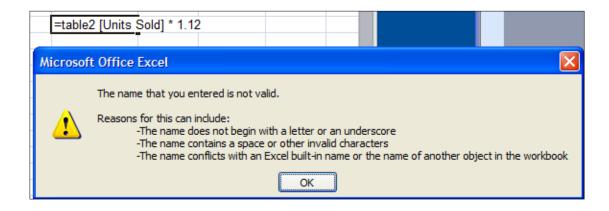
A simpler formula that returns the same result as the one that uses internal names (as long as it's entered in the same row as the Units Sold cell you want to use) is:

=Table2[Units Sold]*1.12

Although the label "Units Sold" is the label for an entire column, in the context of a formula entered in a worksheet cell, Excel operates as if **[Units Sold]** refers to a single cell in that column. That is, the **[#This Row]** designator is assumed even if not entered.

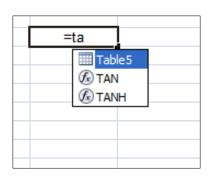
Note

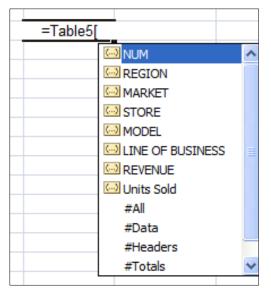
Excel is <u>unforgiving</u> in its requirements for how an internal *table* name reference must be constructed. In the illustration below, a space between the **Table2** name and **[Units Sold]** results in an "invalid name" error message. Eliminate this space and the formula works.



Excel Prompts for Internal *Table* **Names**

If you're constructing a formula by typing and you're using internal Excel *table* names, Excel looks at what you're typing to see if it recognizes what that might be. In the illustration at right, Excel believes that the **=ta** part of the formula might be intended as **=Table5** or as **=TAN** or as **=TANH** (these last two are functions). If more than one option is displayed in the drop-down, select the one you want and hit the TAB key to have Excel complete the typing for you.





In the continuation of the formula entry (illustrated at left), typing an open square bracket ([) results in a display of all the possible internal Excel names that might follow. Notice that this display shows each one of the *table*'s header labels, along with four special qualifiers for *table* elements: #All, #Data, #Headers, and #Totals.

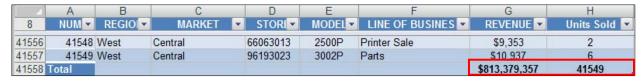
This display puts you a little closer to understanding how Excel names *table* elements. Some experimentation is in order to find out exactly what these names mean.

► Experiment example 1 using the **#Totals** qualifier

- Turn the Total row on in the *table*.
- Enter the formula **=SUM(Table3[#Totals])**
- The formula returns the value 41,549.
- Examine the *table*'s Total row. It appears that the formula is summing any numeric value on that row.



- Change the total row by adding a sum of the values in the **Revenue** column. Go back and check the formula result. It's updated to the value 813,420,906 (or 41,549 + 813,379,357).

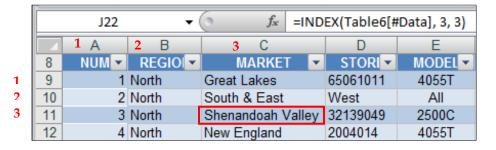


► Experiment example 2 using the **#Headers** qualifier

- In an empty cell in the worksheet enter this formula: =INDEX(Table3[#Headers], 1, 3)
- Note that the syntax for this function is =INDEX(array, row_number, column_number).
- The formula returns the value MARKET. Evidently the virtual name **Table3[#Headers]** references the row of headers as if it's an array. Our formula asked for the first row (there is only one row) and the third column (the column with the header label MARKET).

► Experiment example 3 using the **#Data** qualifier

- In an empty cell in the worksheet enter this formula: =INDEX(Table6[#Data], 3, 3)
- Excel returns "Shenandoah Valley" because it's contained in the third cell in the third row of the array named Table6[#Data].
- We can conclude that the virtual qualifier **#Data** refers to the data in the *table* range but <u>not including</u> the table headers. If **#Data** included the headers, Excel would have returned the Market name "South & East" instead of "Shenandoah Valley".

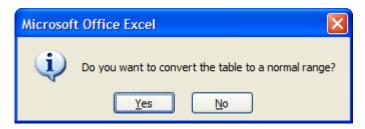


► <u>Use of the #All qualifier</u>

- Based on the behavior of the **#Data** qualifier, one might assume that the **#All** qualifier refers to both the data <u>and</u> the headers in the *table* range.
- I leave it to you to devise an experiment to determine whether or not this is the case.

Convert a Table Back to a List Range

To convert the *table* back to a normal (list) range, choose the "Convert to Range" icon in the "Tools" group on the "Table Tools Design" tab. Excel displays this dialog:



Conversion Effects

If you choose Yes, these things happen:

- If the Autofilter is on (the default for an Excel 2007 *table*), it turns off.
- Any special formatting you added to the table remains.
- Clicking inside the former table no longer displays the "Table Tools Design" tab.
- The internal *table* name Excel assigned the table no longer exists. It no longer displays in the "Name Manager" dialog.
- Any formulas you may have written that used default table or tablecomponent names are converted so the names are eliminated. For example

This formula: =Table2[[#This Row],[Units Sold]]*1.12 is converted to: ='B&E Data'!\$H9*1.12

Confusingly, the formula: **=Table2[Units Sold]*1.12** is in one case converted to the formula **=H9* 1.12** and in another case is converted to this very different formula:

='B&E Data'!\$H\$9:\$H\$41557*1.12

Both formulas return the same result, however.

Convert a Range (that was once a table) Back to a Table

If you now convert the list range <u>back</u> to an Excel 2007 *table*, none of the formulas you wrote when the *table* was a *table* regain the internal Excel names they once used. And, if you recreate one of the sample formulas above using Excel internal names, you'll notice that the name **Table2** no longer exists. For any calculations

with the newly-converted list range, Excel now uses a *table* a name with a number higher in sequence; here, **Table3**.

Miscellaneous Table Notes

When Names are Used/Not Used

If you enter your formula in the same worksheet row that contains the *table* cell your formula references, Excel uses an internal *table* element name when you click the *table* cell as you build the formula. If you enter the formula in a *different* worksheet row from the *table* cell your formula references, Excel does *not* use an internal *table* element name but a regular Excel cell reference. For example, in the first case (same row) a reference might be [#This Row],[UnitsSold] where in the second case (different row) the reference would be H9.

Odd Cell Behavior After Using a Table

If you have been using an Excel 2007 *table* in your worksheet, beware of worksheet cells that seem inoperable. By this, I mean you may try to enter a value or a formula in a cell with no result or an unexpected result. It appears that using worksheet cells for *table* calculations can change the nature of those cells. If you encounter a cell like this, right-click the cell and choose *Clear* from the context menu that displays, even if the cell appears empty. Then try your operation again.

My Conclusion

The Excel 2007 *Table*-Like Range (2003 List)

It appears that confirmed Excel 2003 users have the opportunity to continue using the old "list" range "database" features that were available in 2003. A "normal" list range (not officially designated as a table), for example, works normally with the Autofilter, the Advanced Filter, the dfunctions, and lookup functions. As expected, VLOOKUP and HLOOKUP should be used *without* the list header row while INDEX should be used *with* the list header row. The old 2003 rules about calculating with filtered data continue to apply (filtered, nonvisible cells are included in ranges unless using the SUBTOTAL function).

| | Α | В | С | D | Е | F | G | Н |
|-------|-------|---------|-----------|----------|---------|-------------------|-----------|--------------|
| 8 | NUM 💌 | REGIO[▼ | MARKET - | STOR[🕶 | MODEL ~ | LINE OF BUSINES 🛂 | REVENUE 🛂 | Units Sold 🛂 |
| 2700 | 2692 | South | Illinois | 35034014 | 2500P | Printer Sale | \$89,816 | 23 |
| 5194 | 5186 | South | Illinois | 35034014 | 2500P | Printer Sale | \$81,724 | 26 |
| 11041 | 11033 | South | Indiana | 74076026 | 2500P | Printer Sale | \$82,262 | 26 |
| 13417 | 13409 | South | Tennessee | 23026016 | 2500P | Printer Sale | \$118,819 | 32 |
| 16330 | 16322 | North | Ohio | 34037017 | 2500P | Printer Sale | \$102,670 | 25 |
| 18110 | 18102 | North | Ohio | 34037017 | 2500P | Printer Sale | \$132,727 | 36 |
| 20845 | 20837 | South | Tennessee | 74075025 | 2500P | Printer Sale | \$85,235 | 26 |
| 24954 | 24946 | South | Florida | 63163013 | 2500P | Printer Sale | \$113,619 | 36 |
| 32974 | 32966 | South | Florida | 93097017 | 2500P | Printer Sale | \$93,241 | 26 |
| 41558 | | | | | | | | |

Use of the Term "Array" in 2007

One change necessitated by Excel 2007's new focus on semi-automated Excel database features and its changed use of the term *table* is that "table" can no longer rightly refer to a "list but not including the headers" as it did in Excel 2003. It appears that Microsoft has offered no new term as a replacement for "table". As I read it so far, the old term "array" can still be used, however. Excel 2007's definition for "array" continues to refer two aspects, the second of which can be used as a term for a "list but not including the headers":

- 1) Array: Used to build single formulas that produce multiple results. An array range shares a common formula.
- 2) Array: Operate on a group of arguments that are arranged in rows and columns. An array constant is a group of constants used as an argument.

Use the New Excel 2007 *Table* Features vs. Stick with Excel 2007's 2003-Like "List Range"

Since both options appear to be available in Excel 2007, how do you decide which to use? As mentioned earlier, if you're brand new to Excel you might prefer to handle all Excel database-type operations using the new Excel 2007 *table* features. They're prominent on the ribbons and evidently the way Microsoft expects users to go. On the other hand, if you're an Excel user with experience in the older versions of the application, once you learned how the old, 2003-style "list" worked, list operations were straightforward and relatively uncomplicated.

My own take is that I'll stick with the 03-style list features *unless* there's a compelling reason to use one of the new features offered by the 2007 *table*. Some of the decision points I'll use are summarized below. When I discover more or if now-murky features become more clear to me I'll update this document.

| Excel 2007 Table | Excel 2007 03-"List"-Like Range | Compelling? |
|----------------------------|---------------------------------|-------------|
| Sort | Sort | No |
| Autofilter | Autofilter | No |
| Advanced Filter | Advanced Filter | No |
| Auto-formatting | Manual formatting | Maybe |
| Auto-calculating total row | Manual totals | Maybe |
| Dynamic table expansion | Manual resizing | Maybe |
| Pivot Table available | Pivot Table available | No |
| Text to Columns | Text to Columns | No |
| Remove Duplicates | Remove Duplicates | No |
| Group and Outline | Group and Outline | No |
| Internal (virtual) names | No automatic names | Doubtful |