Data Extraction Techniques for Spreadsheet Records

Volume 2, Number 1
2007
page 119 – 129

Mark G. Simkin
University of Nevada Reno, MarkGSimkin@Yahoo.com, (775) 784-4840

Abstract

Many accounting applications use spreadsheets as repositories of accounting records, and a common requirement is the need to extract specific information from them. This paper describes a number of techniques that accountants can use to perform such tasks directly using common spreadsheet tools. These techniques include (1) simple and advanced filtering techniques, (2) database functions, (3) methods for both simple and stratified sampling, and, (4) tools for finding duplicate or unmatched records.

Keywords
Data extraction techniques, spreadsheet databases, spreadsheet filtering methods, spreadsheet database functions, sampling.

INTRODUCTION

A common use of spreadsheets is as a repository of accounting records (Rose 2007). Examples include employee records, payroll records, inventory records, and customer records. In all such applications, the format is typically the same: the rows of the worksheet contain the information for any one record while the columns contain the data fields for each record. Spreadsheet formatting capabilities encourage such applications inasmuch as they also allow the user to create readable and professional-looking outputs.

A common user requirement of records-based spreadsheets is the need to extract subsets of information from them (Severson 2007). In fact, a survey by the IIA in the U.S. found that "auditor use of data extraction tools has progressively increased over the last 10 years" (Holmes 2002). These data extraction tasks can be as simple as computing an average or total, or as complex as identifying all the records in one worksheet that do not match those in a second worksheet.

Accountants often perform data extraction tasks using database software such as Access or auditing tools such as Audit Control Language (ACL), and using such packages becomes mandatory if the size of the data set exceeds the row capacity of the spreadsheet package at hand. However, if the data sets are smaller than such limits—a common occurrence—there are several reasons why accountants might prefer to use spreadsheets for such extraction tasks instead.

One simple reason is convenience. If the data are already in a spreadsheet file and the user is familiar with extraction methodology, there is no need to export the data to further software in order to perform the desired tasks. A related reason is visibility. Unlike database software, the user is always able to view the records in the dataset at hand—an advantage in that such visibility provides important feedback. Additionally, newer spreadsheet systems now allow users to exploit the foreground or background *colors* of the spreadsheet cells in their work—for example, allow them to sort or select data using *color* as a selection criterion—a capability not available on most older spreadsheet, database, or auditing systems. Finally, using data-extraction techniques in spreadsheets allows users to avoid learning how to use alternate, and perhaps less-convenient, software.

Understanding spreadsheet data-extraction techniques is also useful to accounting *educators*. One advantage is that spreadsheet software is usually available on the computers in college classrooms—a convenience when discussing either the theory or the practice of data-extraction methodology. Another reason is that accounting students are likely to be familiar with some of the spreadsheet skills described here, thus rendering the technical aspects of a given problem relatively manageable. A final advantage stems from the fact that spreadsheet usage is nearly universal in the accounting profession and accounting students can always use these spreadsheet skills to check the work they performed using alternate software.

Excel is but one of many spreadsheet systems that provide tools for performing data-extraction tasks. The purpose of this paper is to discuss such tools in sufficient detail that AIS instructors can use them in their various accounting and auditing classes. For example, the author has successfully used this material in several accounting and IS classes to help students better understand how spreadsheets can be used in place of specialized auditing or database software to perform common data-extraction tasks. The visual aspects of such demonstrations have been especially educational because spreadsheet formulas are self-documenting and the worksheets themselves display immediate results that can be inspected for accuracy and completeness.

The techniques described here can also be reviewed and tested by AIS students directly—a task that competent spreadsheet users should be able to complete in less than an hour. The figures and examples use Excel 2007, although almost all the techniques described here can also be performed using Excel 2003. For illustrative purposes, all the examples in this paper use simplified versions of the spreadsheets required to solve the case in Appendix B of Hunton, Bryant, and Bagranoff (2004). As illustrated in greater detail below, these files are two spreadsheets—one containing the equivalent of a master file of employee records, and one containing the equivalent of a transaction file of payroll records. Copies of these files may be downloaded from the "student resources" section of the Wiley web site for this book at www.wiley.com/college/hunton and are used with the permission of the publisher. Smaller versions of these files are also available directly from the author.

The next section of this paper describes simple and advanced filtering techniques. This section also describes how to use database functions. The second section describes sampling techniques including both simple and stratified sampling of worksheet records. The third section describes techniques for performing record management tasks—in particular, finding duplicate or unmatched records. The final section of this paper summarizes the presentations and discusses some caveats for the work.

SIMPLE AND ADVANCED FILTERING TECHNIQUES

Perhaps the easiest data extraction techniques to understand are those using Excel's filtering tools (Bordelon 2002; Stein 2000). These include the AutoFilter and Advanced Filter tools. A related set of tools are the database functions such as DAVERAGE, which we shall also explore here.

Simple Filters

A common accounting task is to select a subset of records that match a specific criterion. Examples include selecting (1) those customer records with a specific zip code, (2) those product records from a specific vendor, or (3) those sales invoices made on a particular day. All these requirements are easily met using Excel's AutoFilter tool.

To illustrate, consider the Excel worksheet of employee records in Figure 1. Following convention, each row of the spreadsheet contains information about a single employee and each column of the spreadsheet contains the information for a particular data field—e.g., the employee's ID number, name, department code, and so forth. The duplicate records in the set shown in Figure 1 are often errors that you'd like to identify—a skill discussed later in this paper.

Because databases are often quite large, you usually want to view only a subset of the information—for example, to see only those employees working in a particular department or those employees working a standard 40 hours. These are simple tasks to perform with Microsoft Excel's AutoFilter tool, using the following steps:

Step 1: Turn on Excel's AutoFilter. To initialize the AutoFilter tool, locate the cursor anywhere inside the record set and then select Data/Filter from Excel's main menu. This causes Excel to display the small, drop-down arrows located in the headings of the data set (see Figure 1).

Step 2: Select the criteria from a drop down list. The only other step is to select the criteria you wish from the drop down lists provided by the arrows in the headings. For example, to view all those employee records in department 0-2402-000, merely select this value from the drop down list that appears when you click the arrow in Column C (Dept. ID). The result will be a listing of only those employee records satisfying this criterion

It is also easy to select records satisfying multiple criteria. For example, to view those employees who work in department 0-2402-000 *and* who work a standard 40-hour shift, perform the steps above and then click on the value "40" from the drop down list for Column F (Standard Hours). The results are the 14 records of those employees satisfying both requirements.

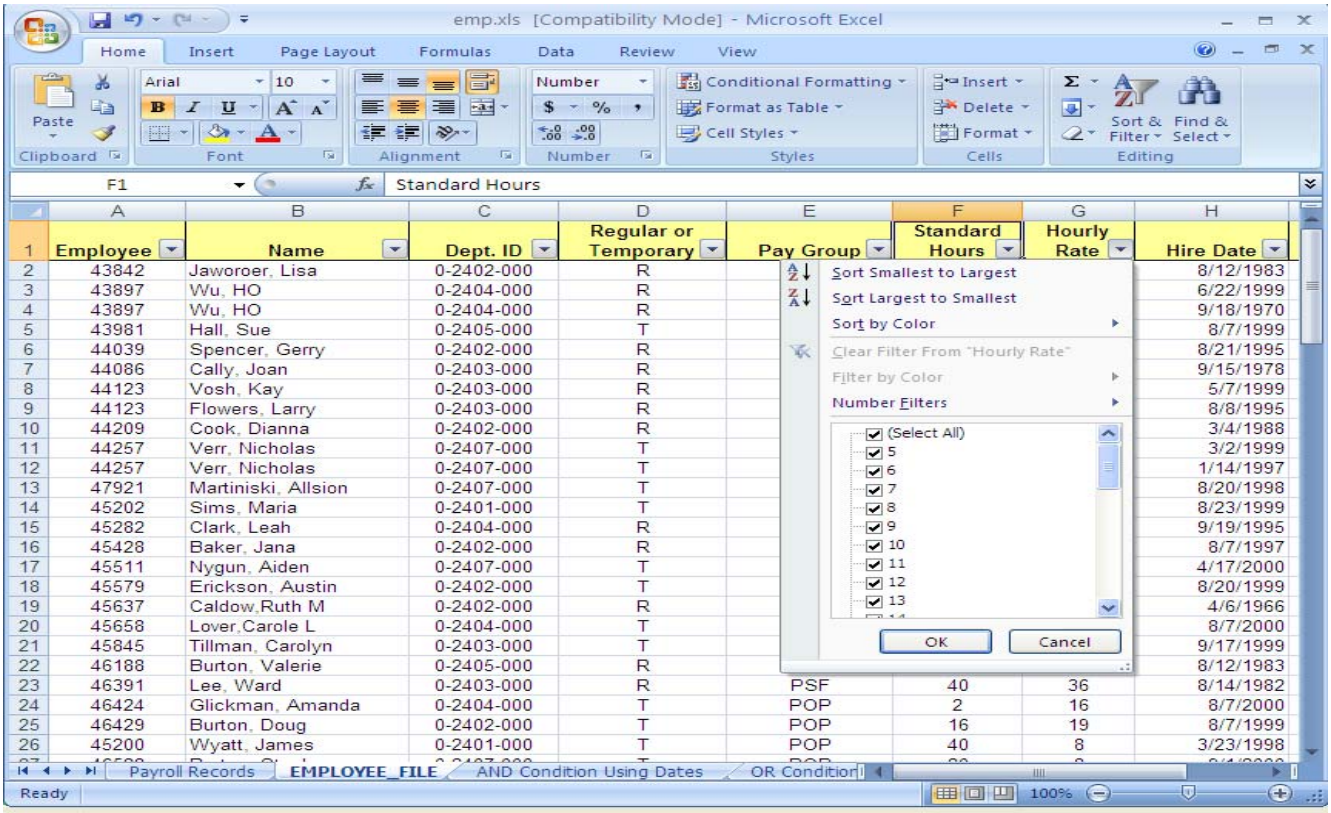
Further AutoFilter Applications

You can continue to add criteria to further filter your list. For example, to view those employees in department 0-2402-0 *and* who have a standard 40-hour shift *and* who have temporary work status, select the corresponding values for the filters in the appropriate columns. As you might imagine, the results will be those employees satisfying all three criteria.

For numerical data fields, you can also perform several additional extraction tasks. For example, if you click on the "Number

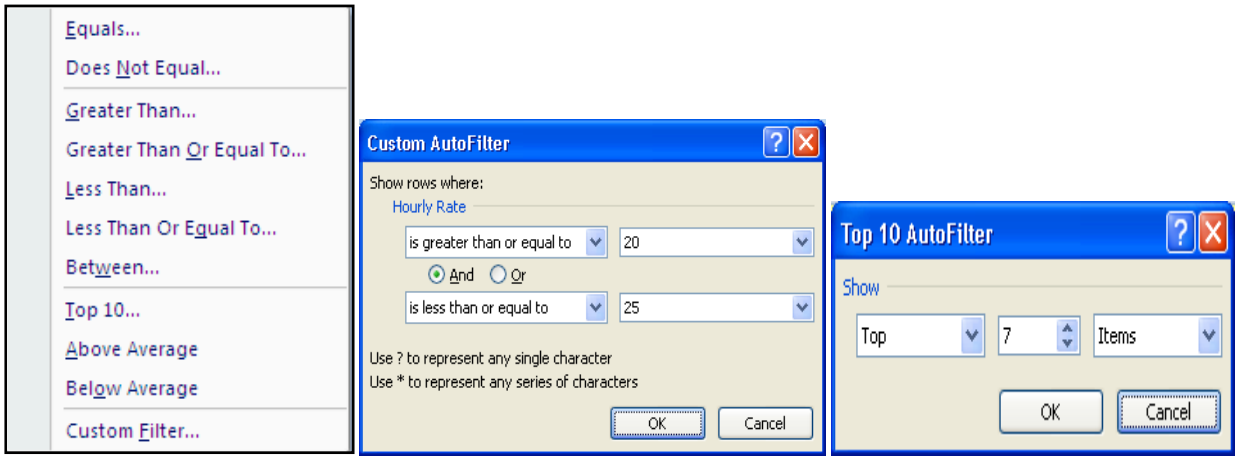
Filters" option in the dialog box of Figure 1, you will see the menu shown in Figure 2(a). Many of these choices are self explanatory—for example, the ability to create filters that select only those employee records with hourly rates greater than, less than, or equal to a value that you specify in a subsequent dialog box, as well as several other choices. If you select "Between," for example, you will see the dialog box shown in Figure 2(b). This example creates a custom filter that limits the record display to employees with hourly rates between $20 and $25.

Figure 1. A set of employee records with arrows created by Excel's AutoFilter and a representative drop-down list for the Standard Hours data field.



Employee	Name	Dept. ID	Regular or Temporary	Pay Group	Standard Hours	Hourly Rate	Hire Date
43842	Jaworoer, Lisa	0-2402-000	R				8/12/1983
43897	Wu, HO	0-2404-000	R				6/22/1999
43897	Wu, HO	0-2404-000	R				9/18/1970
43981	Hall, Sue	0-2405-000	T				8/7/1999
44039	Spencer, Gerry	0-2402-000	R				8/21/1995
44086	Cally, Joan	0-2403-000	R				9/15/1978
44123	Vosh, Kay	0-2403-000	R				5/7/1999
44123	Flowers, Larry	0-2403-000	R				8/8/1995
44209	Cook, Dianna	0-2402-000	R				3/4/1988
44257	Verr, Nicholas	0-2407-000	T				3/2/1999
44257	Verr, Nicholas	0-2407-000	T				1/14/1997
47921	Martiniski, Allsion	0-2407-000	T				8/20/1998
45202	Sims, Maria	0-2401-000	T				8/23/1999
45282	Clark, Leah	0-2404-000	R				9/19/1995
45428	Baker, Jana	0-2402-000	R				8/7/1997
45511	Nygun, Aiden	0-2407-000	T				4/17/2000
45579	Erickson, Austin	0-2402-000	T				8/20/1999
45637	Caldow,Ruth M	0-2402-000	R				4/6/1966
45658	Lover,Carole L	0-2404-000	T				8/7/2000
45845	Tillman, Carolyn	0-2403-000	T				9/17/1999
46188	Burton, Valerie	0-2405-000	R				8/12/1983
46391	Lee, Ward	0-2403-000	R	PSF	40	36	8/14/1982
46424	Glickman, Amanda	0-2404-000	T	POP	2	16	8/7/2000
46429	Burton, Doug	0-2402-000	T	POP	16	19	8/7/1999
45200	Wyatt, James	0-2401-000	T	POP	40	8	3/23/1998

Figure 2. Some examples of Excel's dialog boxes for numeric filters



(a) the menu for number filters **(b) the dialog box for the "Between" numeric filter** **(c) the Top 10 AutoFilter**

New in the 2007 version of Excel is the ability to display records either above or below the average of the column. For example, if you select "Above Average" for the Hourly Rate column in Figure 1, Excel will display the records of only those employees whose pay rates are above the average computed for the entire record set.

Finally, Excel's AutoFilter also provides a "Top 10" option. This feature enables you to select the largest ten values in the column—for example, those employees with the highest hourly rates. If you choose this option from the menu in Figure 2(a), however, Excel will first display the dialog box in Figure 2(c), which enables you to indicate exactly how many such records you'd like to see—for example, "7" in Figure 2(c). Because Excel uses Julian values to represent calendar dates, this "top-ten option" also works on date fields. Thus, if you first format date values to "General" (to see these Julian values) you can then use the AutoFilter feature to select "top ten" for a date column (for example, the Hire Date in Figure 1), Excel will display the ten "largest" values—i.e., the most recent hires. However, if you prefer to continue to display dates in familiar month/day/year formats, it may be easier to obtain such an ordered list by simply sorting the entire record set on this field, from "newest to oldest."

After you have completed a specific filtering operation, you may want to restore your data to their original, viewable state before performing other spreadsheet tasks. To do so, click on the choices Data/Filter from the main menu (i.e., the same choices you made in Step 1 above to engage Excel's AutoFilter in the first place). In other words, the AutoFilter option is a toggle switch. The examples below assume that you restore your data to their initial state before performing a new data-extraction task.

Excel's AutoFilter also provides two sorting options that appear at the very top of the drop down lists—one to sort your records in ascending sequence and one to sort them in descending sequence. You can use these options to sort your entire data set, or to sort only a current subset of records—for example, the top ten wage earners.

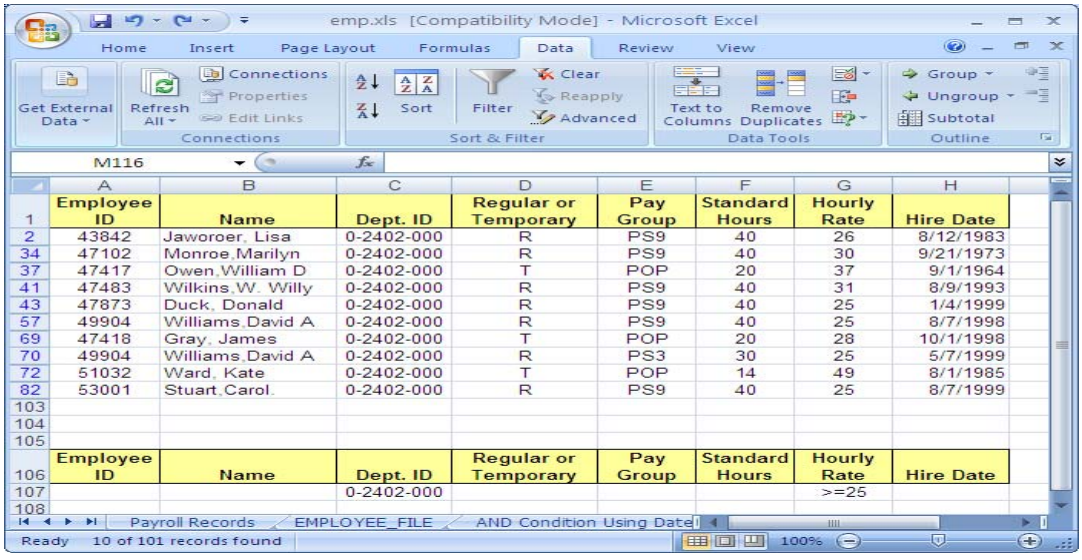
Again, to restore your entire list you can either choose "(All)" from the drop down lists or click again on the Data/Filter icon in the main menu at the top of the screen. The first choice retains the AutoFilter arrows (but restores your view of the entire data set), while the second choice toggles the AutoFilter to "off" position (and also restores your view of the entire data set).

Advanced Filters

Excel's AutoFilter works well for selection criteria involving Boolean And operations, but cannot perform Boolean OR operations—i.e., selecting records that satisfy *either* of two requirements. An example of such an operation would be a listing of those customer invoices that are either less than $100 *or* greater than $500. For these tasks, however, you can create advanced filters, which in fact closely resemble database queries.

To illustrate how to create advanced filters, let us start with a simple example. Suppose you wanted a list of those employees working in department 0-2402-000 whose hourly rates were $25 an hour or more. One solution would be to first filter your list by department 0-2402-000 and then filter your records again on the hourly rate column. An alternate method is to create an advanced filter for this task—a skill that pays dividends when even more complex criteria are involved. To create an advanced filter for those employees working in department 0-2402-000 whose hourly rates are $25 or more, follow these steps:

Figure 3. The criteria range (cells A106:H107) in this spreadsheet specifies those employees who are both in department 0-2404-000 and have hourly rates of $25 or more.



	A	B	C	D	E	F	G	H
1	Employee ID	Name	Dept. ID	Regular or Temporary	Pay Group	Standard Hours	Hourly Rate	Hire Date
2	43842	Jaworoer, Lisa	0-2402-000	R	PS9	40	26	8/12/1983
34	47102	Monroe,Marilyn	0-2402-000	R	PS9	40	30	9/21/1973
37	47417	Owen,William D	0-2402-000	T	POP	20	37	9/1/1964
41	47483	Wilkins,W. Willy	0-2402-000	R	PS9	40	31	8/9/1993
43	47873	Duck, Donald	0-2402-000	R	PS9	40	25	1/4/1999
57	49904	Williams,David A	0-2402-000	R	PS9	40	25	8/7/1998
69	47418	Gray, James	0-2402-000	T	POP	20	28	10/1/1998
70	49904	Williams,David A	0-2402-000	R	PS3	30	25	5/7/1999
72	51032	Ward, Kate	0-2402-000	T	POP	14	49	8/1/1985
82	53001	Stuart,Carol	0-2402-000	R	PS9	40	25	8/7/1999
103								
104								
105								
106	Employee ID	Name	Dept. ID	Regular or Temporary	Pay Group	Standard Hours	Hourly Rate	Hire Date
107			0-2402-000				>=25	
108								

Step 1: Create a criteria range. The criteria range is simply a set of cells with the same cell headings as the initial data set. To avoid misspellings, the author finds it convenient to simply copy the initial headings to a free area of the worksheet. A location *below* the data works fine—for example, cells A106:H107 in Figure 3—but this is not a requirement. (Programming note: you only need the headings for the cells containing criteria entries, not the entire row of headings shown in Figure 3.)

Step 2: Enter the criteria. The requirements of the specific problem dictate what criteria to enter within the criteria range. For this example, you would enter "0-2402-000" in the cell directly under "Dept. ID" and ">=25" in the cell directly under "Hourly Rate," as illustrated in Figure 3. (Programming note: make sure that you do not reverse the order of operators to "=>" when specifying your selection criteria. If you do, Excel will interpret this as an error rather than as a criterion.)

Step 3: Run the Advanced Filter. It now remains to instruct Excel to perform the filtering task required. To do so, select Data/Advanced Filter from the main menu in Excel (or Data/Filter/Advanced Filter using Excel 2003). Excel will display the dialog box in Figure 4 that requires you to enter: (1) the range of cells containing the data, and (2) the range of cells containing the criteria range. For the first item, provide the cell range for the data (including the headings)—e.g., cells A1:H102. For the second item, provide the cell range containing the headings and the selection criteria entries—e.g., cells A106:H107 for the example at hand.

Note that the Advanced Filter dialog box shown in Figure 4 also gives you the choice of filtering the list in place (the current choice here) or copying the filtered list to another location—a handy feature if you want to store a copy of your selected data in another worksheet for further analysis later.

Figure 4. The dialog box for Excel's Advanced Filter option.

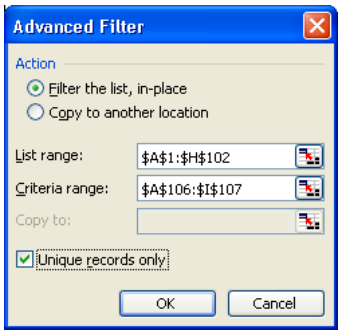


Figure 3 also shows the results of our advanced filter—i.e., those employee records satisfying the requirements specified in the criteria range. But what if you wish to change the criteria in row 107 to examine new extraction requirements? The answer is that Excel will continue to display the prior data and *not* immediately display new results—i.e., the change is not dynamic. Thus, if you wish to see the results for new selection criteria—for example, those employees earning more than $30 an hour—you must run the Advanced Filter again in step 3 above.

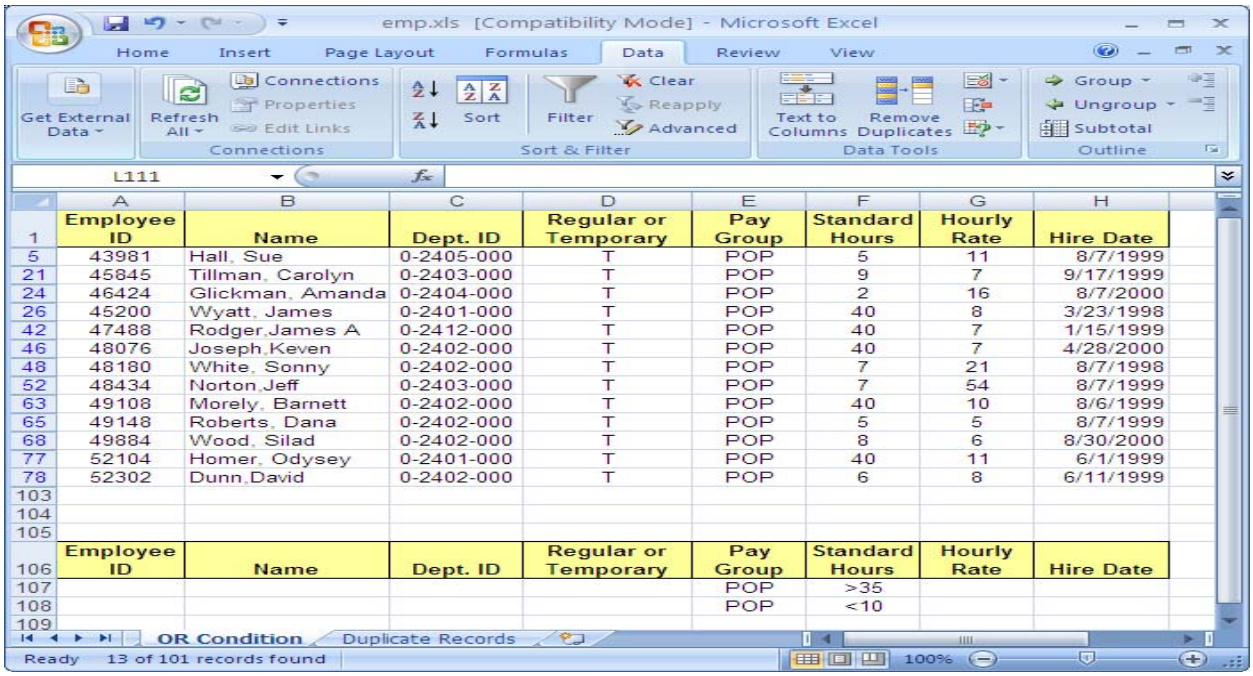
Searching for Blank Fields. It is possible to create many different types of advanced filters. If you wish to search for blanks in a field, for example, just type an equals sign (=) in the criteria range for this purpose. Running the Advanced Filter again will reveal all those records with blanks in the designated field(s).

Wild-Card Searches. Excel provides two symbols for performing wild card searches on text fields—an asterisk (*) and a question mark (?). Use the asterisk as a multi-character wildcard and the question mark for a single wildcard character. For example, to search the employee file for all employees whose last name begins with "W," enter "W*" in the appropriate cell of the criteria range in Figure 3. Similarly, to find those employees beginning with the "Wa," enter "Wa*" in the criteria range. Finally, to find employees whose first and third letters of their last name are "W" and "r," (e.g., Ward or Worthington), enter "W?r" in the criteria range.

Or Searches. Perhaps one of the most useful data extraction tools is an OR filter—i.e., a filter that allows *either* of two conditions to be true. For example, suppose you wanted a list of employees in pay group POP with standard hours *either* greater than 35 or less than 10. You can create such OR selections by placing your criteria on separate lines in the criteria range. Figure 5 provides an example. (Programming note: you will also need to expand the criteria range specified in the dialog box of Figure 4 to include all three rows before running the filter—e.g., to $A$106:$A$108).

Figure 5 also shows the results of this filtering operation—i.e., the (13) records of those employees in Pay Group POP whose standard hours are *either* greater than 35 or less than 10. The "POP" entries in the condition stub limit the search to records in this pay group while the values ">35" and "<10" in the second and third rows of the condition stub express the OR condition desired. (Programming note: it is not necessary to express these filtering conditions directly under the columns containing the data for these conditions. As long as you include the headings, in fact, the entries in cells E106:F108 could just as easily be placed in, say, cells A106:B108.)

Figure 5. The criteria range to select those employees in pay group "POP" with standard hours either more than 35 or less than 10 (bottom portion) and the results of this filtering operation (top portion).



	A	B	C	D	E	F	G	H
1	Employee ID	Name	Dept. ID	Regular or Temporary	Pay Group	Standard Hours	Hourly Rate	Hire Date
5	43981	Hall, Sue	0-2405-000	T	POP	5	11	8/7/1999
21	45845	Tillman, Carolyn	0-2403-000	T	POP	9	7	9/17/1999
24	46424	Glickman, Amanda	0-2404-000	T	POP	2	16	8/7/2000
26	45200	Wyatt, James	0-2401-000	T	POP	40	8	3/23/1998
42	47488	Rodger,James A	0-2412-000	T	POP	40	7	1/15/1999
46	48076	Joseph,Keven	0-2402-000	T	POP	40	7	4/28/2000
48	48180	White, Sonny	0-2402-000	T	POP	7	21	8/7/1998
52	48434	Norton,Jeff	0-2403-000	T	POP	7	54	8/7/1999
63	49108	Morely, Barnett	0-2402-000	T	POP	40	10	8/6/1999
65	49148	Roberts, Dana	0-2402-000	T	POP	5	5	8/7/1999
68	49884	Wood, Silad	0-2402-000	T	POP	8	6	8/30/2000
77	52104	Homer, Odysey	0-2401-000	T	POP	40	11	6/1/1999
78	52302	Dunn,David	0-2402-000	T	POP	6	8	6/11/1999
103								
104								
105								
106	Employee ID	Name	Dept. ID	Regular or Temporary	Pay Group	Standard Hours	Hourly Rate	Hire Date
107					POP	>35		
108					POP	<10		
109								

What if you wanted a list of those employees whose data fell within a range of values—for example, a list of employees whose hourly rates were between $15 and $20 per hour? This is a different requirement than that described immediately above because there are now two conditions that must both be true *for the same data field*. But this task is also easily accomplished using Excel's advanced filtering options. To construct the criteria range, merely enter the column heading ("Hourly Rate") twice in separate cells of the criteria range, as shown in Figure 6. This creates the *And* operation required—i.e., the stipulation that the employee's hourly rate is both greater than $15 *and* less than or equal to $20. The top portion of Figure 7 shows the results. (Programming note: before running this new filter, you will also need to make sure that you reduce the size of the criteria range in Figure 4 to two rows.)

Figure 6. The criteria range to select those employees with hourly rates greater than $15 but less than or equal to $20. Only the cells in the last two columns need be specified in the criteria range of Figure 4.

Employee ID	Name	Dept. ID	Regular or Temporary	Pay Group	Standard Hours	Hourly Rate	Hourly Rate
						>15	<=20

Database Functions

Excel's database functions (e.g., DAVERAGE) allow you to compute conditional values from your record sets based on the selection criteria you specify in a criteria range. Thus, you can use criteria ranges similar to those in Figures 5 and 6 to help you compute values of interest from your record sets. These functions include formulas to compute averages, maximums, minimums, counts, and sums of values in a specific column of your data set, but for only those records satisfying the criteria in your criteria range. To illustrate, suppose you wanted to compute the average, maximum, minimum, and count for the standard hours of those employees earning between $15 and $20. To do so, follow these steps.

Step 1: Create the criteria range. The criteria range for this step is the one shown in Figure 6. The correct entries are also shown in cells G107 and H107 in the lower portion of Figure 7.

Step 2: Create the database formulas. Excel's database functions all have the same general format:

=DFUNCTION(data set range, data field name, criteria range).

DFUNCTION is one of Excel's database functions—e.g., DAVERAGE, DMAX, DMIN, DCOUNT, or DSUM. The *data set range* is exactly that—the range of cells containing your records (including the headings). The *data field name* specification is the title of the column containing the data upon which you wish to perform your computations—for example, "Standard Hours." Finally,

the *criteria range* is the range of cells containing your criteria. Thus, the formula to use in cell G109 to compute the average hourly rate for the employees of interest in our data set is:

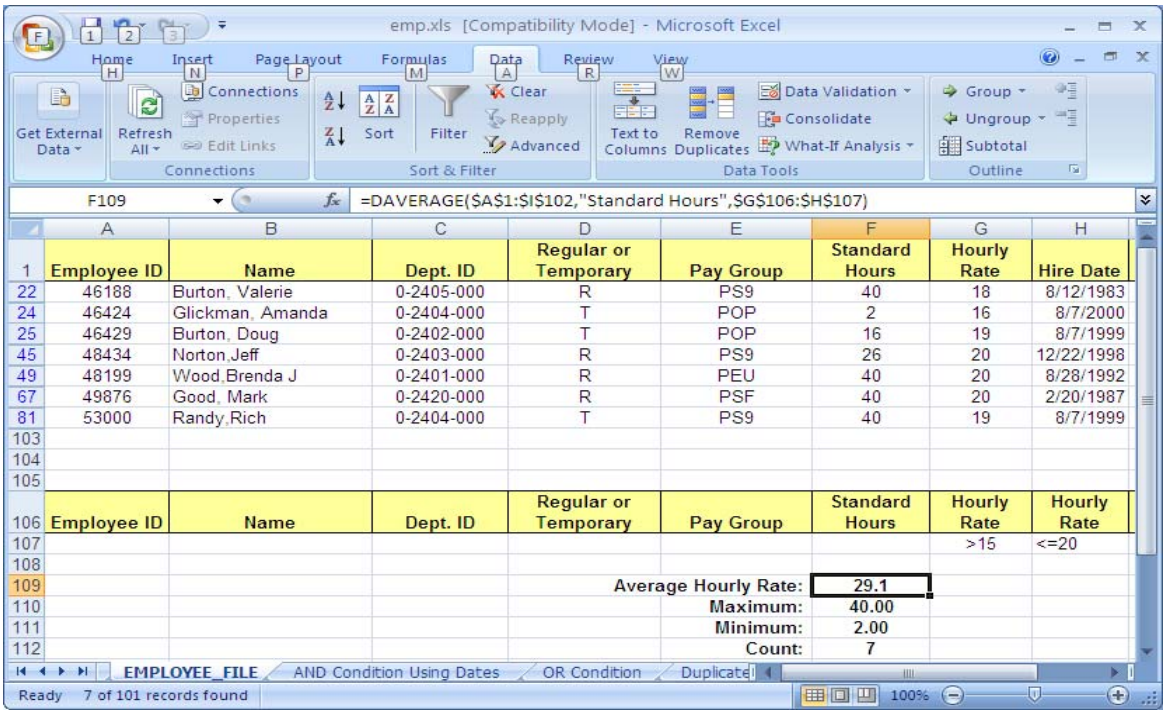
=DAVERAGE($A$1:$I$102,"Standard Hours",$G$106:$H$107)

Figure 7 provides the results for this formula, as well as similar computations for DMAX, DMIN, and DCOUNT. These are the average, maximum, and minimum number of standard hours for those employees earning between $15 and $20 an hour. The count is the actual number of such employees. Because the subset of records is small, you can verify these computed values by inspection in Figure 7. (Programming note: unlike Excel's filtering options, which must be re-executed when you change criteria, Excels' database functions are dynamic. This means that they will automatically re-compute new values whenever new criteria are entered into the criteria range or when new values are inserted into the underlying data set.)

SAMPLING

When the number of records in a spreadsheet becomes large and detailed analysis is required, a common practice is to select a subset of them for examination. For example, an auditor might wish to select a small number of customer records to verify their current account balances. This section of the paper examines two such data-extraction tasks: (1) simple sampling and (2) stratified sampling.

Figure 7. An example of Excel's database functions. All the formulas in this example reference the same criteria range in cells A106:H107.



	A	B	C	D	E	F	G	H
1	Employee ID	Name	Dept. ID	Regular or Temporary	Pay Group	Standard Hours	Hourly Rate	Hire Date
22	46188	Burton, Valerie	0-2405-000	R	PS9	40	18	8/12/1983
24	46424	Glickman, Amanda	0-2404-000	T	POP	2	16	8/7/2000
25	46429	Burton, Doug	0-2402-000	T	POP	16	19	8/7/1999
45	48434	Norton,Jeff	0-2403-000	R	PS9	26	20	12/22/1998
49	48199	Wood,Brenda J	0-2401-000	R	PEU	40	20	8/28/1992
67	49876	Good, Mark	0-2420-000	R	PSF	40	20	2/20/1987
81	53000	Randy,Rich	0-2404-000	T	PS9	40	19	8/7/1999
103								
104								
105								
106	Employee ID	Name	Dept. ID	Regular or Temporary	Pay Group	Standard Hours	Hourly Rate	Hourly Rate
107							>15	<=20
108								
109					Average Hourly Rate:	29.1		
110					Maximum:	40.00		
111					Minimum:	2.00		
112					Count:	7		

Simple Sampling

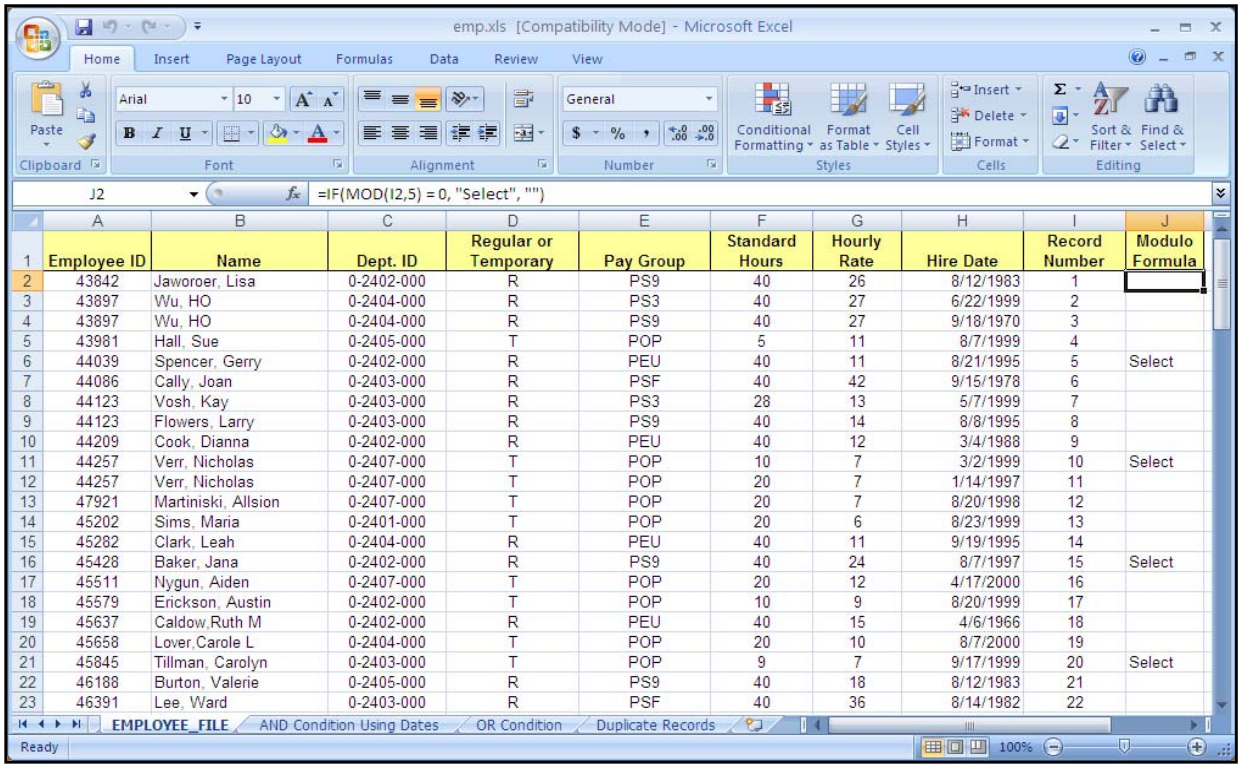
One sampling technique is to select every *nth* record in a record set. For example, suppose you wanted to select every 5th employee record in our illustration for your sample. To perform this sampling task, follow these steps.

Step 1: Add two additional columns to the record set. You will need two additional columns in your record set to perform this task as shown in Figure 8. The first column simply numbers the records consecutively as shown. (Programming note: If you begin the number sequence 1, 2, 3 in the first few cells, you can highlight these cells and then use Excel's data fill handle and a mouse drag to create the remaining values quickly.)

Step 2: Create an IF test using the Mod Function. Use the second new column for the selection process. In particular, we now wish to select records 5, 10, 15, and so forth—i.e., records whose newly-assigned record numbers are evenly divisible by 5. The Mod function, which returns the remainder from a division, can help with this task. In Excel, this function is written as Mod(X, Y), where

X is the starting number and Y is the divisor. For example, Mod(23, 5) is 3 because 5 divides into 23 four times with a remainder of 3.

Figure 8. A method for selecting every fifth record for a sample.



	A	B	C	D	E	F	G	H	I	J
1	Employee ID	Name	Dept. ID	Regular or Temporary	Pay Group	Standard Hours	Hourly Rate	Hire Date	Record Number	Modulo Formula
2	43842	Jaworoer, Lisa	0-2402-000	R	PS9	40	26	8/12/1983	1	
3	43897	Wu, HO	0-2404-000	R	PS3	40	27	6/22/1999	2	
4	43897	Wu, HO	0-2404-000	R	PS9	40	27	9/18/1970	3	
5	43981	Hall, Sue	0-2405-000	T	POP	5	11	8/7/1999	4	
6	44039	Spencer, Gerry	0-2402-000	R	PEU	40	11	8/21/1995	5	Select
7	44086	Cally, Joan	0-2403-000	R	PSF	40	42	9/15/1978	6	
8	44123	Vosh, Kay	0-2403-000	R	PS3	28	13	5/7/1999	7	
9	44123	Flowers, Larry	0-2403-000	R	PS9	40	14	8/8/1995	8	
10	44209	Cook, Dianna	0-2402-000	R	PEU	40	12	3/4/1988	9	
11	44257	Verr, Nicholas	0-2407-000	T	POP	10	7	3/2/1999	10	Select
12	44257	Verr, Nicholas	0-2407-000	T	POP	20	7	1/14/1997	11	
13	47921	Martiniski, Allsion	0-2407-000	T	POP	20	7	8/20/1998	12	
14	45202	Sims, Maria	0-2401-000	T	POP	20	6	8/23/1999	13	
15	45282	Clark, Leah	0-2404-000	R	PEU	40	11	9/19/1995	14	
16	45428	Baker, Jana	0-2402-000	R	PS9	40	24	8/7/1997	15	Select
17	45511	Nygun, Aiden	0-2407-000	T	POP	20	12	4/17/2000	16	
18	45579	Erickson, Austin	0-2402-000	T	POP	10	9	8/20/1999	17	
19	45637	Caldow,Ruth M	0-2402-000	R	PEU	40	15	4/6/1966	18	
20	45658	Lover,Carole L	0-2404-000	T	POP	20	10	8/7/2000	19	
21	45845	Tillman, Carolyn	0-2403-000	T	POP	9	7	9/17/1999	20	Select
22	46188	Burton, Valerie	0-2405-000	R	PS9	40	18	8/12/1983	21	
23	46391	Lee, Ward	0-2403-000	R	PSF	40	36	8/14/1982	22	

For the application at hand, we seek those records for which 5 divides our new record numbers evenly. This is equivalent to finding those values for which Mod(X, 5) = 0, where X is the record number. In cell J2 of the spreadsheet, we would therefore create the following IF test:

=IF(MOD(I2, 5) = 0, "Select", "")

This formula will display the word "Select" if the remainder from the implied division is 0 (i.e., the record number is evenly divisible by 5) and nothing otherwise. The choice of what to display if the IF test triggers true—for example, the word "Select" here—is arbitrary. (Programming note: To select every *nth* record instead of every fifth record, simply replace the number "5" in this IF formula with the value for "n"—e.g., "10" or "20.")

Step 3: Filter the List. You can now use Excel's AutoFilter tool to perform the final selection process. Simply turn on the AutoFilter and choose the word "Select" from the drop down list for column J. In this example, the result is every fifth record from our original list—i.e., the desired sample—which you can then copy elsewhere for further analysis.

Stratified Sampling

The term "stratified sampling" refers to the simple idea of selecting a proportional number of observations from each distinct subset of a population. In taking a sample of vendor invoices, for example, you would want to make sure you selected invoices from many different vendors and when taking a sample of corporate purchases, you would want to include purchases from all the purchasing agents. For our employee file illustration, we might want to ensure that we chose a proportional number of regular and temporary employees, or that we selected both part time and full time workers.

Stratified sampling in Excel is a simple matter of sorting the records in your record set *before* taking every 5th or 10th one of them. Because the records are grouped first by the category of interest, selecting every nth record ensures that a proportional number of them will be selected for the sample. To illustrate, suppose that you wanted to make sure that your sample contained employees from each pay group. To accomplish this, first sort the records by pay group using Excel's Data/Sort option. Then perform the same steps outlined above. As long as there are more than five members in each pay group, your random sample will contain a representative number of them—i.e., will be a stratified sample.

FINDING DUPLICATE AND UNMATCHED RECORDS

Another common data-extraction task is to identify duplicate records in a record set. In a payroll master file, for example, we would expect to see only one record per employee—duplicate records would not make sense. A related task is finding unmatched records. For example, in a payroll application, we might want to look for those employees who received paychecks but did not have matching master file records. This section of the paper describes how to perform both tasks with spreadsheet tools.

Finding Duplicate Records

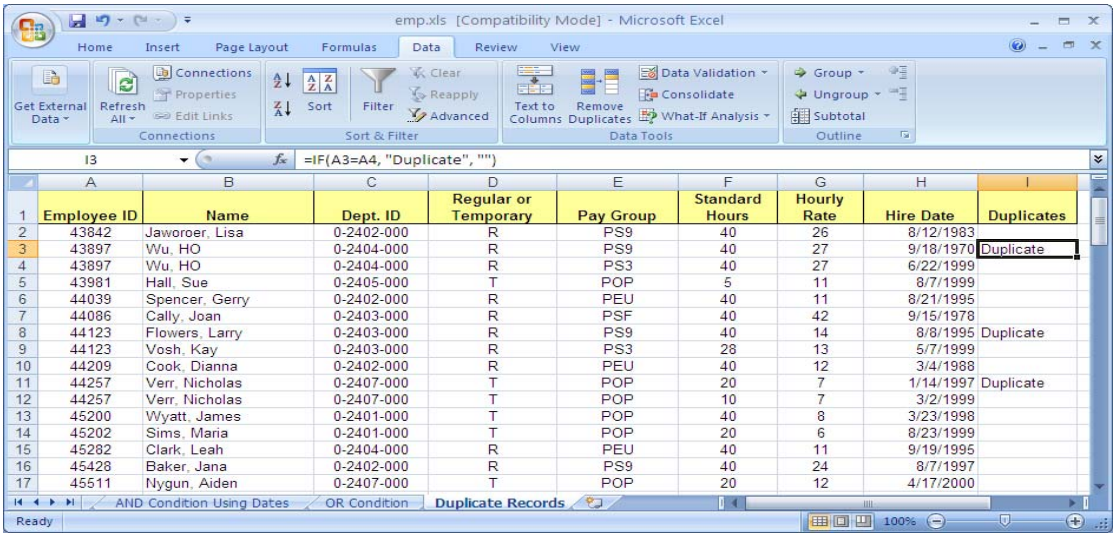
Although Excel does not provide a menu tool for identifying duplicate records, it is a straightforward matter to create a simple formula to accomplish this task. Suppose, for example, we want to find all the duplicate records in our employee file (Figure 1). To do so, follow these steps:

Step 1: Sort the file by primary key. More than one employee might have the same name, but presumably, the Employee ID codes are unique. Your first step, therefore, is to ensure that the records are in sequential order by this code. Sort the records accordingly—i.e., by Employee ID number. Figure 9 is a partial listing of the results.

Step 2: Create a "Duplicates" Formula. Add a new column to your spreadsheet as illustrated in Figure 9. The figure shows this additional column at the far right side of the data set, but this location is arbitrary. In your new column, create an IF test that displays something (e.g., the word "Duplicate") if the Employee ID number in the current row matches the number directly below it, and nothing otherwise. For example, the formula for cell I2 is:

=IF(A2=A3, "Duplicate", "").

Figure 9. Using an IF test to identify duplicates in spreadsheet records.



	A	B	C	D	E	F	G	H	I
1	Employee ID	Name	Dept. ID	Regular or Temporary	Pay Group	Standard Hours	Hourly Rate	Hire Date	Duplicates
2	43842	Jaworoer, Lisa	0-2402-000	R	PS9	40	26	8/12/1983	
3	43897	Wu, HO	0-2404-000	R	PS9	40	27	9/18/1970	Duplicate
4	43897	Wu, HO	0-2404-000	R	PS3	40	27	6/22/1999	
5	43981	Hall, Sue	0-2405-000	T	POP	5	11	8/7/1999	
6	44039	Spencer, Gerry	0-2402-000	R	PEU	40	11	8/21/1995	
7	44086	Cally, Joan	0-2403-000	R	PSF	40	42	9/15/1978	
8	44123	Flowers, Larry	0-2403-000	R	PS9	40	14	8/8/1995	Duplicate
9	44123	Vosh, Kay	0-2403-000	R	PS3	28	13	5/7/1999	
10	44209	Cook, Dianna	0-2402-000	R	PEU	40	12	3/4/1988	
11	44257	Verr, Nicholas	0-2407-000	T	POP	20	7	1/14/1997	Duplicate
12	44257	Verr, Nicholas	0-2407-000	T	POP	10	7	3/2/1999	
13	45200	Wyatt, James	0-2401-000	T	POP	40	8	3/23/1998	
14	45202	Sims, Maria	0-2401-000	T	POP	20	6	8/23/1999	
15	45282	Clark, Leah	0-2404-000	R	PEU	40	11	9/19/1995	
16	45428	Baker, Jana	0-2402-000	R	PS9	40	24	8/7/1997	
17	45511	Nygun, Aiden	0-2407-000	T	POP	20	12	4/17/2000	

Copy this formula to all the remaining cells in your new column. The result is a column containing blanks for non-duplicate records and the word "Duplicate" for those records with matching records below them (see again Figure 9). Note that this technique will also identify records with 2 or more duplicates as well as a single repetition.

Step 3: Filter the List. It is now a simple matter to use Excel's AutoFilter tool to limit the display to the duplicate records. To do so, turn on Excel's AutoFilter and select the word "Duplicate" from the drop down list. The result is the list of duplicate records, which can also be copied to an alternate worksheet for further analysis if desired.

Finding Unmatched Records

A seemingly difficult task for spreadsheet users is to find those records in one file that do not match records in a second file. But even this task can be accomplished easily with Excel tools by treating the second file as a large lookup table. We begin our task by examining the file containing the initial data. This is often a transaction file that you want to match against a master file. Figure 10 provides an example—an employee payroll file for a particular period (this is a simplified version of the employee payments file that is also available from the Wiley web site). To find which records in this file have no matching records in the master file in Figure 1, follow these steps.

Step 1: Copy the transaction file into a separate worksheet of the master file's workbook. This step is technically not necessary, but it makes the subsequent programming easier and is therefore recommended. As when finding duplicate records, it is also important that the records in the master file be in ascending, record-key sequence. For the example at hand, this means in ascending order of employee ID. This sorting requirement is *not* necessary for the records of the transaction file, however.

Step 2: Add a column to the transaction file that contains a VLOOKUP formula. For the current illustration, this new column was added to the far right side of the spreadsheet set, but this is arbitrary. In this column, we wish to create VLOOKUP formulas that treat the entire set of records in the master file as one large lookup table. The general form of the lookup function is:

= VLOOKUP(lookup value, table range, column offset, Boolean value)

In this formula, the *lookup value* is typically a reference to the cell containing a value you want to find in your table. In most unmatched-record applications, you will use the foreign key for this—i.e., the Employee ID field. The *table range* is the range of cells containing the lookup table itself. In many alternate spreadsheet applications, this is a small table but in unmatched-records applications such as this one, it will be the entire set of employee master-file records that are stored in another worksheet. The *column offset* indicates the column number within the table you wish Excel to display. In this application, the actual value is not important because what you really care about is whether Excel can *find* a matching record—not the values in the records it does find. Arbitrarily, the current example will try to display the employee's name.

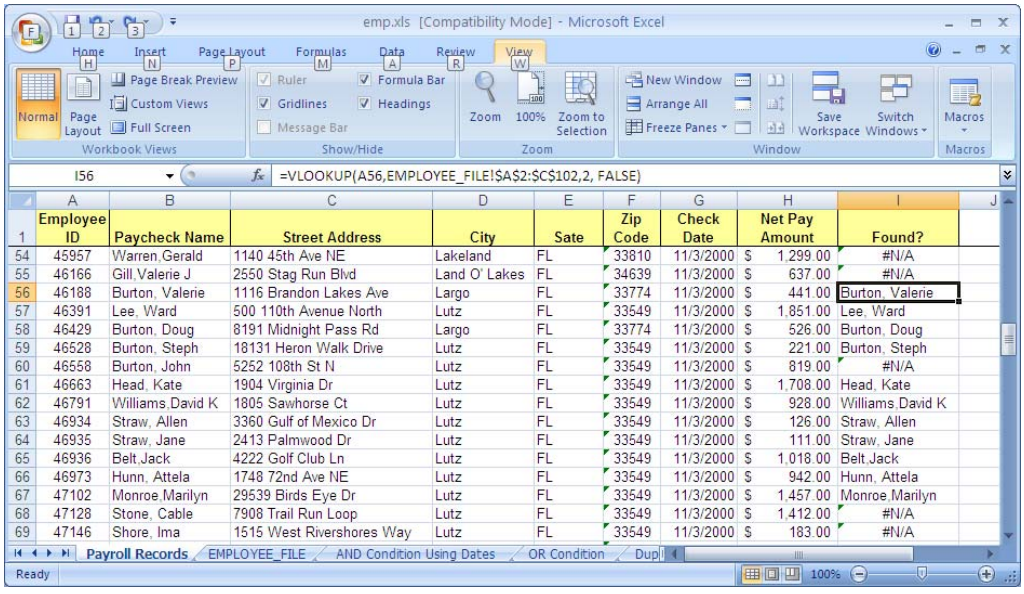
The last argument in the VLOOKUP function, *Boolean value*, is typically omitted in spreadsheet applications. This parameter indicates whether or not you want to limit your lookup formula to exact matches. Setting this value to TRUE means that you don't want to limit to exact matches (the default), while setting this value to FALSE means that you do. We want to limit our return values to exact matches so we must explicitly set it to FALSE. A representative formula for cell I2 is therefore:

=VLOOKUP(A2, EMPLOYEE_FILE!$A$2:$C$102, 2, FALSE)

In this formula, the value to lookup is in cell A2 (i.e., the employee ID number), the lookup table is the entire data set in the Employee master file (i.e., cells A2:C102 of the EMPLOYEE_FILE worksheet), the value to display is the employee's name (which has a column offset of "2"—refer back to Figure 1), and the FALSE setting indicates that we only want exact matches.

Step 3: Use the AutoFilter to display unfound records. Figure 10 provides a partial listing of the results so far. For records that Excel can find, we see the employee's name in our new column—i.e., the names in column I that were found in the master file. For records that Excel cannot find, we see the error message "#N/A," indicating an unfound record condition. To limit the list to these unmatched records, use Excel's AutoFilter and select "#N/A" from the drop down list (it typically appears at the very bottom). You can then copy this final list elsewhere if you want to analyze the results further.

Figure 10. A portion of a payroll (transaction) file that you wish compare to the master file in Figure 1.



	A	B	C	D	E	F	G	H	I
1	Employee ID	Paycheck Name	Street Address	City	Sate	Zip Code	Check Date	Net Pay Amount	Found?
54	45957	Warren,Gerald	1140 45th Ave NE	Lakeland	FL	33810	11/3/2000	$ 1,299.00	#N/A
55	46166	Gill,Valerie J	2550 Stag Run Blvd	Land O' Lakes	FL	34639	11/3/2000	$ 637.00	#N/A
56	46188	Burton, Valerie	1116 Brandon Lakes Ave	Largo	FL	33774	11/3/2000	$ 441.00	Burton, Valerie
57	46391	Lee, Ward	500 110th Avenue North	Lutz	FL	33549	11/3/2000	$ 1,851.00	Lee, Ward
58	46429	Burton, Doug	8191 Midnight Pass Rd	Largo	FL	33774	11/3/2000	$ 526.00	Burton, Doug
59	46528	Burton, Steph	18131 Heron Walk Drive	Lutz	FL	33549	11/3/2000	$ 221.00	Burton, Steph
60	46558	Burton, John	5252 108th St N	Lutz	FL	33549	11/3/2000	$ 819.00	#N/A
61	46663	Head, Kate	1904 Virginia Dr	Lutz	FL	33549	11/3/2000	$ 1,708.00	Head, Kate
62	46791	Williams,David K	1805 Sawhorse Ct	Lutz	FL	33549	11/3/2000	$ 928.00	Williams,David K
63	46934	Straw, Allen	3360 Gulf of Mexico Dr	Lutz	FL	33549	11/3/2000	$ 126.00	Straw, Allen
64	46935	Straw, Jane	2413 Palmwood Dr	Lutz	FL	33549	11/3/2000	$ 111.00	Straw, Jane
65	46936	Belt,Jack	4222 Golf Club Ln	Lutz	FL	33549	11/3/2000	$ 1,018.00	Belt,Jack
66	46973	Hunn, Attela	1748 72nd Ave NE	Lutz	FL	33549	11/3/2000	$ 942.00	Hunn, Attela
67	47102	Monroe,Marilyn	29539 Birds Eye Dr	Lutz	FL	33549	11/3/2000	$ 1,457.00	Monroe,Marilyn
68	47128	Stone, Cable	7908 Trail Run Loop	Lutz	FL	33549	11/3/2000	$ 1,412.00	#N/A
69	47146	Shore, Ima	1515 West Rivershores Way	Lutz	FL	33549	11/3/2000	$ 183.00	#N/A

SUMMARY AND CONCLUSION

Many spreadsheets store accounting records in them. When the size of such record sets is large, it is convenient to employ data extraction tools, rather than manual inspection, to find specific information. This paper demonstrated several such techniques, including (1) simple filtering techniques using Excel's AutoFilter, (2) advanced filtering techniques using Excel's Advanced Filter, (3) database functions, and (4) methods for taking both simple and stratified samples of data from large data sets. This paper also discussed two additional data extraction techniques: (1) a method for finding duplicate records in a data set, and (2) a method for identifying unmatched records using two data sets.

There are other software packages that also perform the data-extraction tasks discussed here, and the author recognizes that there might be important reasons why an accounting professional might prefer, or be required, to use them. Among them are: (1) conformance to corporate guidelines, (2) preferences for alternate software, (3) the need to perform yet additional data-manipulation tasks that spreadsheets cannot easily accomplish, or (4) preferences for alternate methodology. On the other hand, using spreadsheet tools for such extraction tasks provides a convenient approach to select needed data in an efficient, convenient, and often highly-visual manner.

REFERENCES

Bordelon, P. L. 2002. Targeting Spreadsheet Data. *Journal of Accountancy* 193(6): 60-63.

Holmes, B. 2002. It's in There Somewhere. *Accountancy* 129(1306): 61-62.

Hunton, J. E., S. M. Bryant, and N. A. Bagranoff 2004. *Core Concepts of Information Technology Auditing* New Jersey: John Wiley and Sons.

Rose , J. 2007. Taking Human Error out of Financial Spreadsheets. *Strategic Finance* 88(9): 53-33.

Severson, J. 2007. Navigate Speedily in Excel Data. *Journal of Accountancy* 203(1): 56-57.

Stein, J. D. 2000. Spreadsheet Smarts. *Journal of Accountancy* 189(1): 53-60.