ListRx

from

TalkBiz
DOT COM
Hard Core How-To For Business

Disclaimer: I have tested this software extensively and it has performed as designed in each case. **That does not mean it's foolproof**. There's no way to predict or test its interaction with all the possible hardware and software configurations on Windows machines, or the unusual uses to which some people may put the program.

I also can’t guarantee that you won't make mistakes. You should always make backup copies of important files before making changes to them.

Use of this software is entirely at your own risk.

This software is solely for your own use. It may not be re-sold, given away, shared or otherwise distributed by anyone but TalkBiz, Inc.

That specifically means that you *do not* have resale or reprint rights to any part of the ListRx package!

Sorry. It had to be said.

ListRx Documentation

ListRx: The Basics

Don't let the simple look and small file size of ListRx fool you. This software, especially in conjunction with Tagger (included with the package) and your own email-merge program, makes options available to you that no other desktop software ever has.

The interface lets you set up each operation in a simple process that's easy to understand. Take the time to read this documentation and the accompanying manual, and you'll be amazed at the powerful options you discover for increasing your profits.

The first thing to understand about ListRx is that the software is designed in a "Wizard" format. That is, it lets you set up each operation in logical steps, and allows you to move backward and forward through the steps to change or repeat various parts of the process.

The Wizard consists of four screens:

In the first screen, you choose the file(s) to work with, and tell the program what format they use and how to handle the input data.

The second screen lets you tell the program what you want it to do with the data.

The third is where you tell the program how you want the output data formatted, and where to store it.

The fourth is where you tell it "Yeah. I'm ready. Do it." The software keeps you informed about where it is in the process of following your instructions, and lets you know when it's done.

"Here's what I've got. Here's what I want. Here's how and where I want it. Do it."

It's that easy.

There are three main things to keep in mind while using ListRx. The first:

Always work with a COPY of your database!

Once you've successfully completed your operations, you can copy the new files over your originals if that's what you want. However, things can always go wrong, so I recommend working with a copy - every time.

While I haven't been able to get this to crash, and haven't seen any bugs that could cause data damage, it's not possible to predict all the potential conflicts with the various software and hardware configurations in use at any one time.

A far larger potential for problems comes from simple human error. ListRx offers the option of deleting entries from your source lists and/or databases. If you make a mistake with a filter, especially while using wildcards, it's easy to fix - if you don't make it on the original.

Your lists and databases are among the most valuable assets your business has. Don't take risks with them.

The second is less scary, but something you need to understand to see the full power of the program. It's simply that ALL the specified operations you tell the program to perform are cumulative.

That is, if you tell it to extract all the entries with AOL addresses from your database, randomize them, remove any that match your filters and save them to files with 1000 entries each, it will do that all in one operation.

You'll see how powerful this is as you become more familiar with the program.

To protect against mistakes and potential data loss, and to make it easy to recover from errors, I've added a backup feature to ListRx. Any time the program is told to perform an operation that deletes data from the original file, it will first make a backup copy of that file.

If the original file is named Original.txt, the name of the backup file will be in the format:

Original.txt-DATE-TIME.bak

The date is expressed in the order Year, Month and Date. The time is in military format. So, if you made a change to a file on May 7th, 2004, at exactly 4:36 PM, the backup file name would be

Original.txt-040507-163600-1.bak

This backup will be in the same directory as the original.

To go back to the original file in case of mistakes, you would simply rename or delete the newly modified version and rename the backup to Original.txt.

NOTE: You still need to work from copies!

For one thing, Tagger does NOT have this backup feature. More significantly, it's a very useful habit to get into. Believe me, you'll thank me for insisting on this if you extend this habit to your work with other files.

The third thing to keep in mind will seem like a stumbling block to some. That's simply that **the program only works with text lists and tab- or comma-delimited databases.**

It will not work directly with databases in Access, Approach or other proprietary formats.

This is not a problem. Every database program will allow you to export your databases in those formats, and re-import the modified databases into their proprietary formats.

Just use the "Save as" or "Export as" function, choose tab-delimited, and you're ready to go. I recommend using tab-delimited for one very basic reason: Much of the data used by online businesses originates in web form submissions. It's possible (and not that uncommon) to enter quotes and commas in web forms.

It is not possible to enter a tab character into one.

The reasons for choosing text-delimited formats over proprietary ones were compelling. First, many lists and databases are already stored in text format. Many email-merge and list management programs will only work with files in these formats.

Most importantly, it ensures that anyone can use this software, without the need for fancy add-ons or expensive or complicated conversion tools.

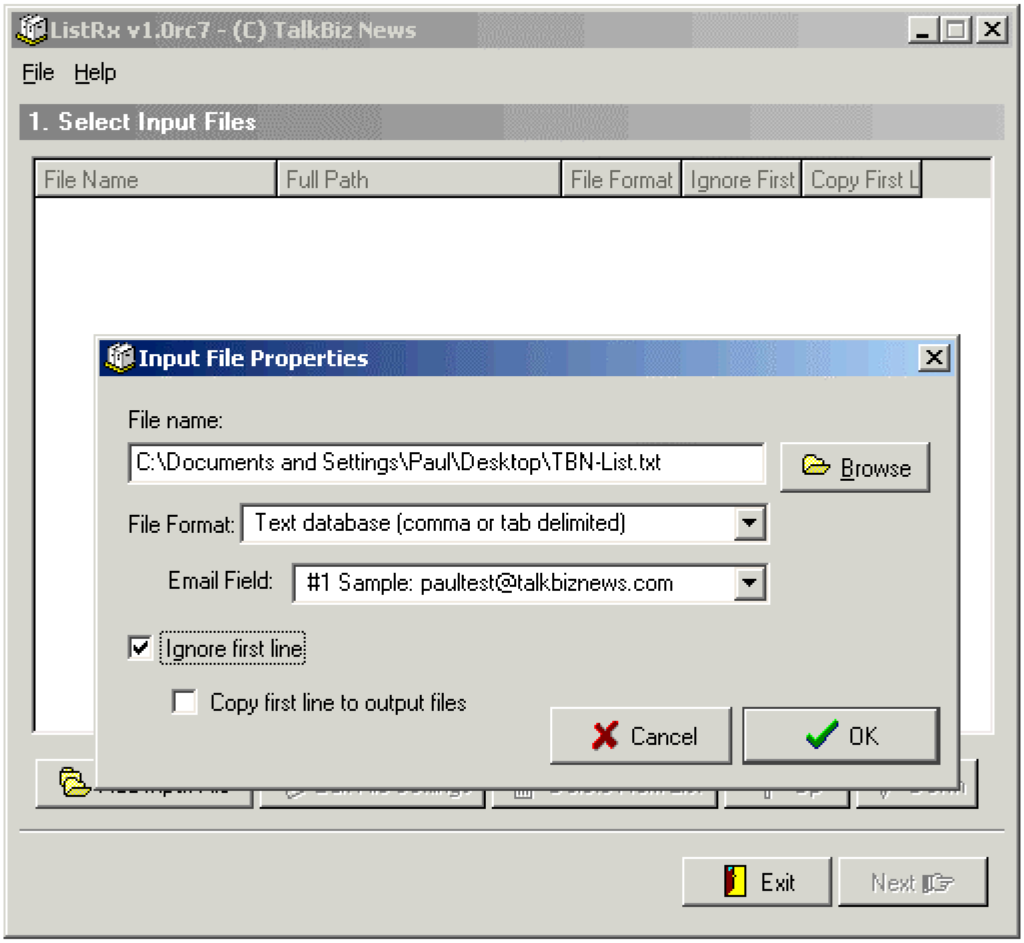
This documentation will explain the technical usage of the ListRx software. The marketing manual that is included with the package tells you how to use the software to boost your business.

Be sure to read this manual and practice with copies of your lists or databases as you go. It's a very simple to use program, but there are a fair number of options, and understanding how they work together will go a long way toward making sure you get maximum results from your investment.

Okay. Let's get started.

Setting Up The Operation

Add Input File: Clicking this button will bring up a screen that looks like this:



File name: C:\Documents and Settings\Paul\Desktop\TBN-List.txt

File Format: Text database (comma or tab delimited)

Email Field: #1 Sample: paultest@talkbiznews.com

☑ Ignore first line

☐ Copy first line to output files

On this screen you have a number of options. The first is the Format of the input file. The choices are:

Plain List (Just emails)

This should be chosen if the file consists of a plain list of just email addresses, one per line. Something like this:

```
Paul@TalkBiz.com
anonymous@example.com
freddie@just-business.com
DaPrez@whitehouse.gov
```

Plain List (Contains emails)

In this case, the list would be in the format:

```
Paul Myers <paul@talkbiz.com>
John Doe <anonymous@example.com>
Freddie Kruger <freddie@just-business.com>
Abe Lincoln <DaPrez@whitehouse.gov>
```

In this case, the program will read the characters between the < and > as the email address.

If you want to extract just the email addresses from a list like this, you can do that using the “Save only emails” function shown later in this documentation.

Text Database (Comma or Tab-delimited)

A Tab-delimited database (with field names on the first line) would be in a format like this:

```
First   Last    Email
Paul    Myers paul@talkbiz.com
John    Doe     anonymous@example.com
Freddie         Kruger          freddie@just-business.com
Abraham         Lincoln         DaPrez@whitehouse.gov>
```

Comma-delimited could be in either of the following formats:

```
First,Last,Email
Paul,Myers,paul@talkbiz.com
John,Doe,anonymous@example.com
Freddie,Kruger,freddie@just-business.com
Abraham,Lincoln,DaPrez@whitehouse.gov
```

Or quoted comma separated values…

```
"First","Last","Email"
"Paul","Myers","paul@talkbiz.com"
"John","Doe","anonymous@example.com"
"Freddie","Kruger","freddie@just-business.com"
"Abraham","Lincoln","DaPrez@whitehouse.gov"
```

These are called CSV files, for Comma Separated Values.

If you're working with a file that uses a different delimiter, such as a | (called a "pipe"), you have a couple of options. The easiest is to treat it as a comma- or tab-delimited file and use the Substrings filtering option, as described later.

A more comprehensive way would be to use a text editor to replace all instances of the delimiter with a Tab character. To do this, load the file into any text editor and use the Find/Replace function. In the Find section, enter the delimiter. In this instance, the | character.

In the Replace section, type in ^T. That's a shifted 6 (on most keyboards) and the capital T. This will put a tab character wherever the | exists in the original database.

This should only be a consideration if you're importing data from an unusual program. Some web-based systems use the pipe character, for example. Most standard database programs will allow you to export the data in tab- or comma-delimited format.

Email Field

If your database has multiple fields, you'll be able to choose the Email field from this drop down menu.

In the rest of this documentation, we'll call this the "Target" field. It was called the Email field in the interface because the vast majority of people using it will filter on that exclusively.

Note that you can actually filter on any target field. This is handy if you're exporting entries from a database based on, for example, the zip codes of customers, or which product(s) they've ordered.

Ignore First Line (checkbox)

If the input file has the field names in the first line, you'll want this box checked. This keeps the program from altering the field names in the input file or moving this line to the output file, unless the next option is checked.

Copy First Line To Output File (checkbox - only available if above is checked)

If you want to use the output file in a database program or mail merge software, you'll want this box checked, so the program knows what each field is called.

Edit File Settings

Clicking on this button will bring up the same screen as above, allowing you to adjust the settings for your source file(s).

There may be times when you want to do multiple operations on the same input file, using different fields as the target or "Email" field.

Delete From List

Clicking on this will remove any highlighted file(s) from the Input Files list.

Up/Down Arrows

Highlighting one of the input files and using these arrows will change the order of the files. This is useful when you want the first line (field names) or entire entries (as explained on the next screen) copied to the output file.

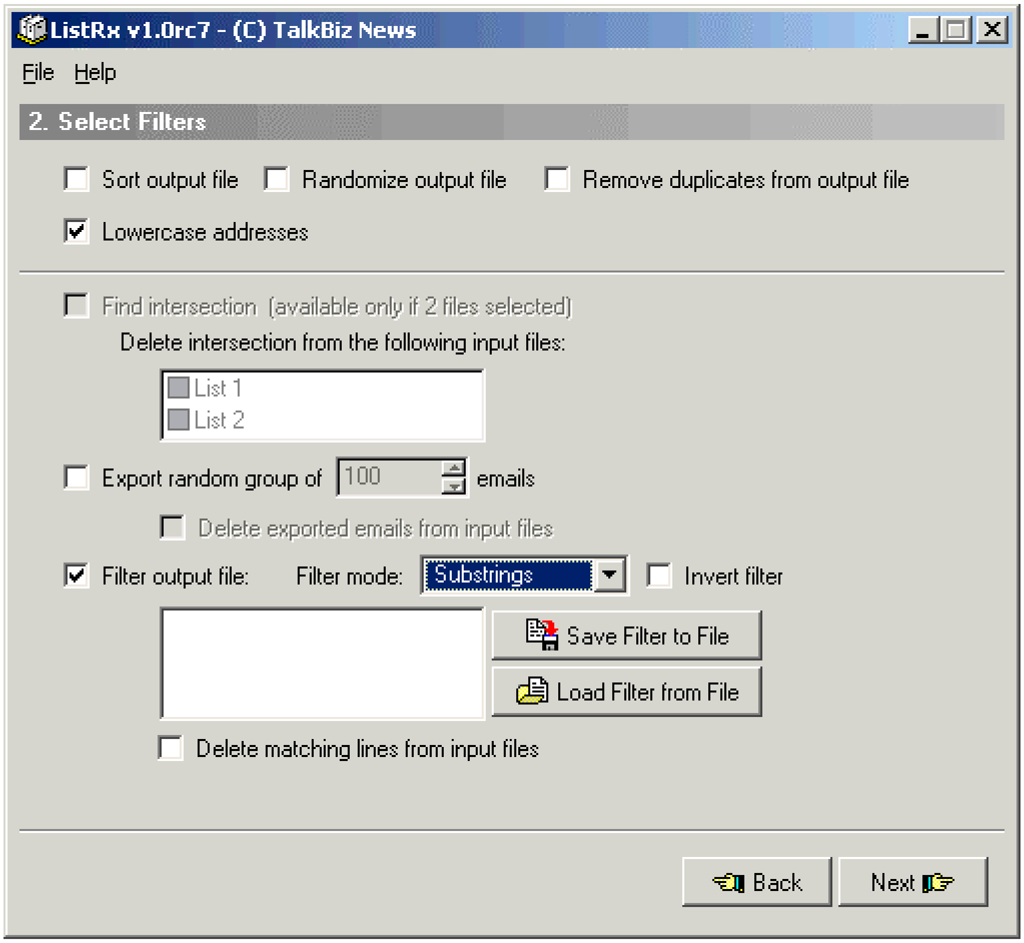
The first file in the list, in this case, will be the one the entries are copied from.

Exit

Closes the program. You can also close it by clicking on the X in the upper right hand corner of the program screen.

Next

Move on to the next screen, Select Filters, which looks like this:



ListRx v1.0rc7 - (C) TalkBiz News

2. Select Filters

☐ Sort output file ☐ Randomize output file ☐ Remove duplicates from output file

☑ Lowercase addresses

☐ Find intersection (available only if 2 files selected)

Delete intersection from the following input files:

☐ List 1
☐ List 2

☐ Export random group of 100 emails

☐ Delete exported emails from input files

☑ Filter output file: Filter mode: Substrings ☐ Invert filter

☐ Delete matching lines from input files

Sort Output File

If this box is checked, the entries in the output file will be sorted based on the contents of the target (called "Email") field. If left unchecked, they'll remain in the same order in which they occur in the input file(s)

Randomize Output File

If this box is checked, the entries in the output file(s) will be randomly ordered, based on the contents of the target field. This is extremely important for purposes of testing, as explained in the companion manuals.

Remove Duplicates From Output File

This will remove duplicates that occur in the input file(s) before they're copied to the output. The first field containing a specific entry in the target field will be left as is. Any other entry containing the same data in the target field will be removed before writing the output file.

Note: In most cases, you should only use this option if you're filtering based on data that should be unique to one individual or company, such as email addresses, phone numbers, or entire lines of text that are treated as one "field."

Lowercase Addresses

This option is checked by default. As a rule, you'll almost always want to leave it checked. The only exception is when you're certain that the entries you're filtering on are exactly the same, case and all, as in the target field.

This treats the contents of the target field as all lower case while it does the required filtering. It does NOT change the case that's written to the output file. That stays the same as it appeared in the input file(s).

Find Intersection

This only applies if there are multiple input files selected. It will find all instances of the Target field that are the same in ALL of the input files.

The program will, unless you check the "Save Only Emails" option on screen three, output the entire matching line as it appears in the first file in the input files list.

Delete Intersection From The Following Input Files

If you want the common entries to be removed from one or more the original source files, check the box next to each one you want "purged" this way.

Export Random Group Of [ ] Emails

This will pull whatever number of entries you specify randomly from the input file(s) and copy them to a separate file. This is very handy for things like surveys and tests sent to limited numbers of people.

Delete Exported Emails From Input Files

If this box is checked, the entries included in the random sample will also be deleted from the input file(s), allowing you to test different things and avoid later sending the same email to folks who've already received it.

Filter Output File: (checkbox)

If this box is checked, the input file(s) will be filtered based on a match for the various criteria you set. To select the degree of matching required, select one of the following from the Filter Mode drop down menu.

Substrings

If this is selected, any entry that contains the string(s) in the filter will be considered a match. For email lists, this is most useful when you want to filter based on domains. For example, with this checked, you might enter…

@hotmail.com
@yahoo.com

… into the text box where the filters are displayed. (To the left of the buttons for "Save Filter to File" and "Load Filter from File.")

In this case, every address that ends with @yahoo.com or @hotmail.com will be matched.

This could also be used to filter addresses that contain words or strings of characters you might now want to allow on your list. Examples might be asdf and asdg. These are often used when some… person… thinks that forging a fake address into a form is in his best interest.

You might have a list of vulgarities that you don't want to allow. Or email addresses or domains that are regularly used to hassle you. Or countries that you prefer not to send your newsletter to. Or autoresponder service domains. Or... whatever else you wish to keep out of your database.

Wildcards

Here we get into more complicated pattern matching, but not excessively so.

In the Substring example, you're looking for the full string anywhere in the target field. With pattern matching, you can get more control over the criteria for a match.

The wildcard character is simply a *. It stands for any or no characters in addition to the specified patterns. For example, the pattern paul*.com would match any email address that began with paul and ended with .com.

paul.com would match mrpaul@example.com, paul1234@talkbiz.com, and gerry@pauls.com

When using this option, make sure that you look over the resulting output file to make sure you haven't cast too wide or too narrow a net with your pattern.

Exact Match

The contents of the target field must match one or more of the list of filters EXACTLY. The only exception occurs if you leave the "Lowercase emails" option turned on and the filter terms are all lowercase.

This is handy if, for example, you're using a list of email addresses that your system logged as wanting to unsubscribe as your filter. Or if you're filtering on a list of phone numbers.

Invert Filter

If this box is checked, the program will act on any target field that does NOT match the filter criteria.

An example: Suppose you want to send an email to everyone in your database who is NOT at a free email service, and you have a list of those domains, one per line, that you're going to use as a filter. You'd check the "Invert Filter" box, load the filter from disk (where it's stored as a text file), and NOT delete the matching entries from the input file.

If you wanted to send a different email to the addresses at free email services, you'd go back and change the name of the output file, uncheck the "Invert Filter" box, and run the program again.

One with and one without the free service addresses.

Text Box For Filter

This is where your filter criteria will be displayed. If you're running a one time filter, you can type the criteria for a match into this box, one per line.

Save Filter To File

If you want to save the contents of the filter box to a new file for later use, click this button and save as you would in any program. I suggest using .flt as the file extension, so you can keep the filters separate and make sure their use is obvious.

You can use any extension you like, however. If you create many filters in a text editor, you may want to create a directory called Filters, and store them there as .txt files.

Load Filter From File

Clicking this button will load a text file into the filter box, one line per filter. You can use as many pre-set filters as you like, but you can only load one file at a time. Attempting to load a second will clear the first from the filter box.

If you have multiple filters you want to use in one operation, you'll need to use a text editor to combine them.

This is NOT a bug. It's intended to prevent the accidental use of extra filters, and to make it easier to clear the box if you load the wrong filter by mistake.

Note: For filter files over 5,000 entries, your computer may slow down while the file loads. This should only happen for a few seconds. I recommend against attempting to load filter files of more than 10,000 lines.

Most people will never need a filter anywhere near that big.

If you have filter files with a LOT of criteria, it's probably best to do the filtering in stages, using 3000 or so filter lines per stage.

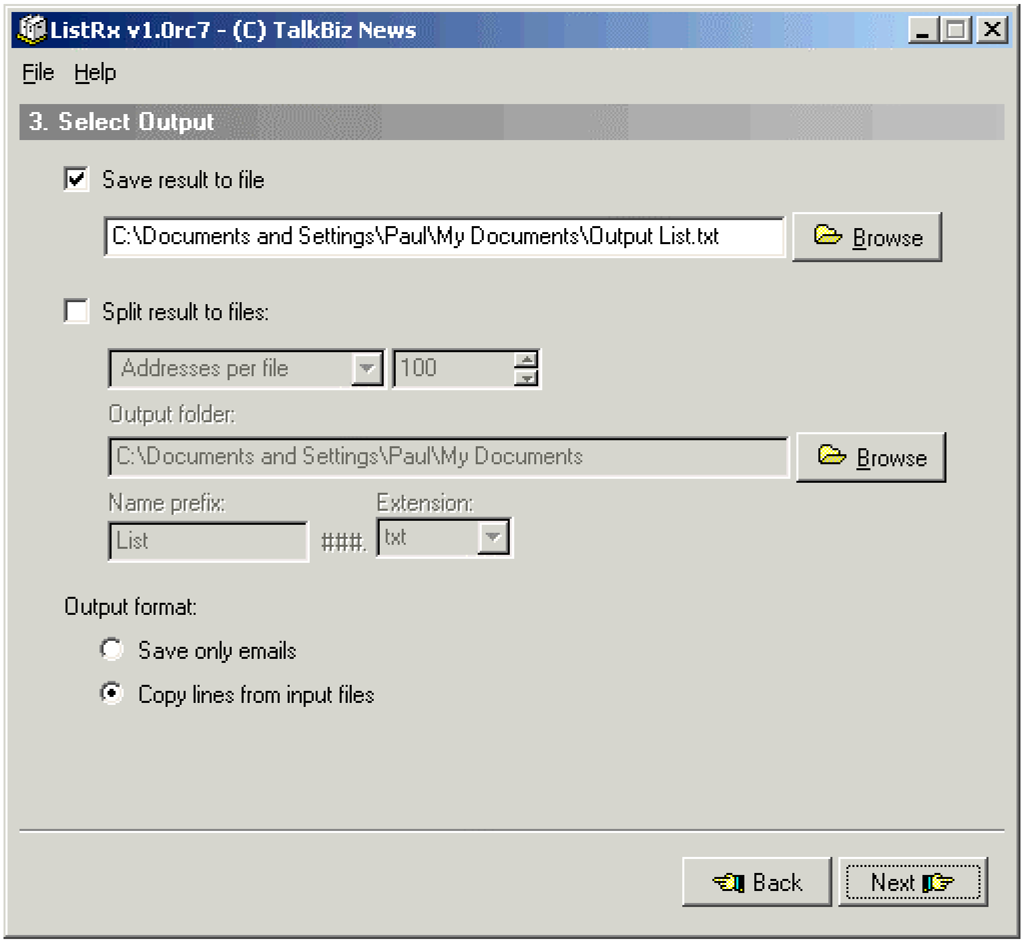
If you're removing email addresses and have the addresses to be removed in a file, one per line, it's faster to use the Intersection function, rather than the filter function.

To do this, add the main database as the first input file, and select the Email field. Then add the file containing the email addresses. Select the option for finding the intersection, and check the box to remove duplicates from the first file, but not the second.

Delete Matching Lines From Input File(s)

This tells the program to delete any line in the input file where the target field contains a match for the filtering criteria.

Click on Next to go to screen three...



☑ Save result to file

C:\Documents and Settings\Paul\My Documents\Output List.txt

□ Split result to files:

Addresses per file 100

Output folder:

C:\Documents and Settings\Paul\My Documents

Name prefix: List ###.

Extension: txt

Output format:

○ Save only emails

● Copy lines from input files

Save Result To File: (checked by default) "My Documents\Output List.txt" [Browse]

You can leave this as is, or use the Browse button to choose another directory and file name for the output file.

Split Result To Files: (checkbox)

If checked, this allows you to split the output data to multiple files. Note that you CAN leave the "Save results to file" and "Split result to files" options both checked.

If you do this, I recommend that you do all filtering FIRST, and then randomize and split the resulting output file.

The drop down menu offers the following choices:

Addresses Per File

This will split the results to whatever number of files are necessary, with the specified number of addresses in each, and the remainder (if it's not an equal number) in the last file. You can change the number of addresses by using the up and down arrows, or by highlighting the number that appears in the box by default and typing in the desired number.

Number Of Files

This will split the resulting data into a specified number of files, with as close to an equal number of entries as possible in each file.

As example of where this would be helpful: You want to know what day of the week is best for you to send your mailings to a specific list. You'd randomize and split the database into 7 equal parts, and send the exact same email to each list, one on each day of the week.

The ONLY difference in the emails themselves would be something in the URL that would allow you to track which response came from each mailing. You don't want to assume that every click that came in on Tuesday was from Tuesday's mailing, for example.

Output Folder

This tells the program what directory to put the split files into as it does its work.

Name: Prefix

The beginning of each file name, which will be followed by sequential numbers, ending with 001, 002, etc.

Extension: txt lst csv db dat

This is the file extension you want appended to each split file. .txt can be used for any output file.

.lst is best used for files with just the email address alone on each line.

.csv is best for comma-delimited files, as some programs handle them automatically when they're named properly.

.db and .dat are appropriate for use with programs that will recognize those extensions and when the data output is in the correct format for those program/extension combinations.

So, if you choose to break the output into 7 parts, the prefix is Day and the extension is .txt, you'll end up with:

```
Day001.txt
Day002.txt
Day003.txt
Day004.txt
Day005.txt
Day006.txt
Day007.txt
```

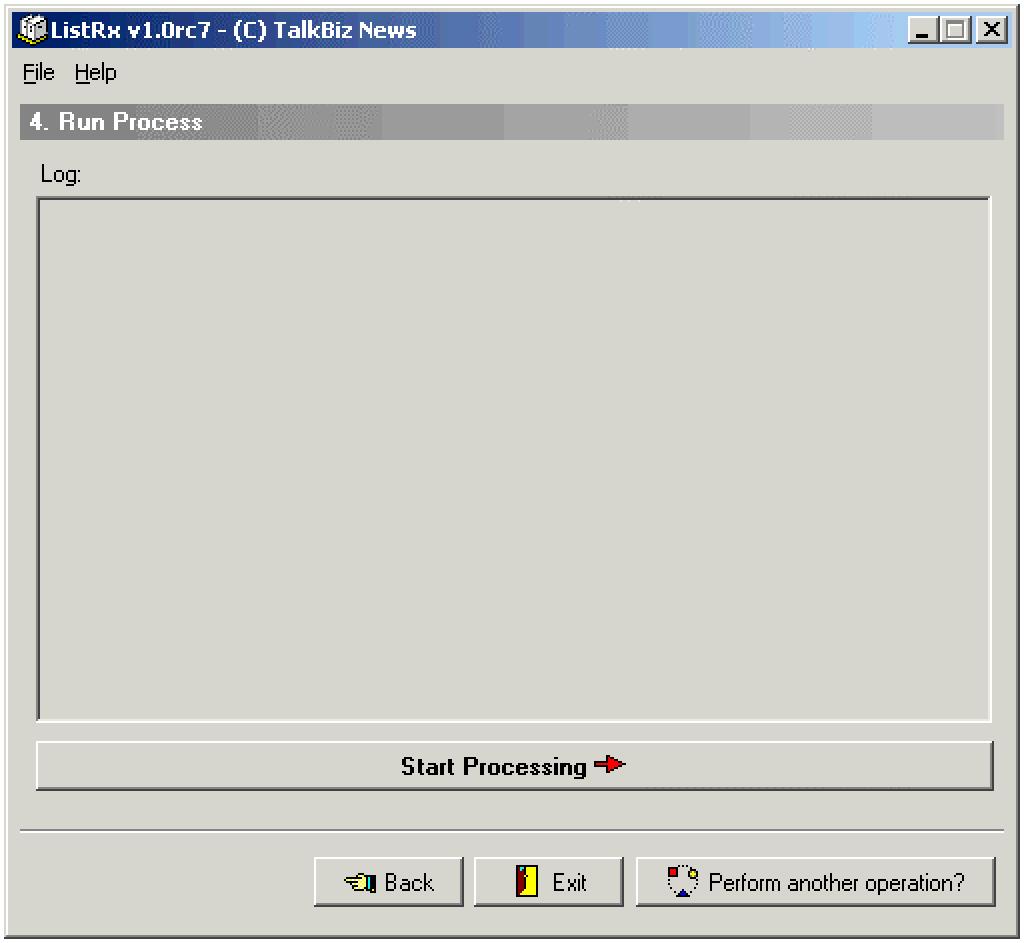
Save Only Emails

This saves only the contents of the target field to the output file, one per line.

Copy Lines From Input Files

This will copy the entire line in which the target field matches the filter criteria. This is checked by default, since most databases contain more than just the email address, and you'll probably want to use the recipient's name or other data in the emails sent.

Now you're ready for screen four...



Start Processing

Clicking on this tells the program that you're ready to go. It will show you where it is in each step of the process, until it's completed.

When it's done, you can choose on of the following:

Back

Back up through the screens to correct a setting, if you goofed, or to perform another operation on the original input file. Remember: it may have been altered by the first operation!

Exit

This shuts down the program.

Perform Another Operation?

Takes you back to the first screen.