

T/MAKER

Vol. 1, No. 3, May, 1982

USERS' GROUP

NEWSLETTER

T/MAKER NEWS

The latest version of T/Maker available is 2.6. This release adds a new function called "DATA" which lets you specify a default data drive to be used for all file names given without a disk drive prefix. In other words, if you say "DATA B", then the file name "XYZ" would really mean "B:XYZ". A default value can be set with T/Modify. The function is very handy if you work with the T/Maker programs on one disk and all your data on another.

Also added in 2.6 are a number of shorthand techniques for underlining or bolding entire lines, and an "All" option to the Tally function which causes it to tally all the lines in a file and not just those which have a plus sign in the first column.

The next release of T/Maker will be 3.0 which will include completely revised Align and Print functions. The result will be a competitive word processing system with many new features. This version will probably not be available until October and will include all new documentation, hopefully more oriented towards the beginning user. T/Maker III will be available as an update to current users which means it will cost about \$50 for a new manual and set of programs. There is a modest but nonzero probability that T/MUG may offer updates in a couple of formats at a reduced price.

Incidentally, this newsletter is being done with a test version of T/Maker III on an Epsom printer, so you may notice some new things like indented paragraphs, underscoring, and **boldface**. All done with ease.

If you have anything special on your "want list" that does not require scrapping all of T/Maker as is, now might be a good time to send in your suggestions.

TABLE OF CONTENTS

Use of LOAD/UNLOAD	2
Keystroke macro:	
Putting lists side-by-side	5
Typing the envelope	6
Answer to Walter E. Lewis	7
Opening an IRA	9
Percents, percents, and more percents	10

USE OF LOAD/UNLOAD

Submitted by Stuart Domber

The use of LOAD/UNLOAD commands require two files. A data file and a mask file. Values in the data file are LOADED into their appropriate locations in the mask file by simply placing the name of each value in the desired mask location inclosed between two braces. Likewise, values can be UNLOADED from a table or text file to form a data file according to a format specified by a mask file.

The following is a practical application demonstrating the use of the LOAD function. The application consists of a data file named VALUES and a mask file named PORTFOLIO (see figures below). Notice how the LOAD command, along with some others, were placed in the top line of the mask file to form a macro or program.

Operation of this module consists of only two steps;

1. GET PORTFOLIO. This command brings mask PORTFOLIO into work area.
2. DO. This command removes the first line of the mask and executes it. This line calls for LOADING the VALUES file into the mask, COMPUTES the mask's equations, CLEANS the work file, and then PRINTs the PORTFOLIO APPRAISAL report.

:	DATE = 02/28/82	:
:	NEW YORK STOCK EXCHANGE	:
:	ABC = 24.75 1.2	:
:	DEF = 25.625 2.2	:
:	GHI = 25.125 1.6	:
:	JKL = 27.00 2.68	:
:	MNO = 41.75 .93	:
:	PQR = 16.375 .16	:
:	STU = 19.75 .44	:
:	VWX = 10.00 .6	:
:	YZ = 29.50 1.92	:
:	AMERICAN STOCK EXCHANGE	:
:	AAA = 11.375 .30	:
:	OVER THE COUNTER	:
:	AAAA = 2.125 .10	:
:	BBBB = 32.813 0	:
:	CCCC = 98.00 10.00	:
:	CASH = 1 0	:
:		:

file VALUES

LOAD VALUES COMPUTE CLEAN PRINT 0 132

PORTFOLIO APPRAISAL

(DATE)

STUART DOMBER

	SECURITY NAME	TICKER SYMBOL	COST	NO OF SHARES	UNIT COST	UNIT VALUE	MARKET VALUE	UNREALIZED GAIN--LOSS	% DIF	DIV/ INT.	ANNUAL INCOME
EX			999,999.99	999999	999.999	999.999	999,999.99	999,999.99	999.9	99.99	9999.99
ZV			-	-	-	-	-	-	-		
AC1			+	/	=						
AC2				+		\$	=				
AC3			-			+	=				
AC4			/				+	%			
AC5				+						\$	=

+	AMERICAN BAGLE	ABC	16712.50	600		{ ABC }				{ }	
+	DELTA ECOLOGY FUND	DEF	497.00	200		{ DEF }				{ }	
+	CASH RESERVE MGT	CASH	1250.00	1250		{ CASH }				{ }	
+	GENERAL HOTEL IND.	GHI	17600.00	600		{ GHI }				{ }	
+	JAMAICA KILD LTD	JKL	14650.00	1200		{ JKL }				{ }	
+	MAINE NOME ORCH.	MNO	13562.50	500		{ MNO }				{ }	
+	PRINTING QUICK RE.	PQRC	6875.00	500		{ PQR }				{ }	
+	STAR TAX UNITED	STU	12712.50	300		{ STU }				{ }	
+	VICTORY WATER EXCH	VWX	6075.00	200		{ VWX }				{ }	
+	YELLOW ZONE FARMS	YZ	10562.50	500		{ YZ }				{ }	
+	ALL AMERICAN AGRI.	AAAA	10000.00	4000		{ AAAA }				{ }	
+	ALFA ASTRO ARMS	AAA	8000.00	1000		{ AAA }				{ }	
+	BETTER BUY BONDS	BBB	25000.00	1000		{ BBBB }				{ }	
+	COUPON CAMERA CO	CCC	20000.00	200		{ CCCC }				{ }	

CC											
EX			999,999.99	,	,	,	999,999.99	999,999.99	999.9	,	99,999.99
AC1			/					+	%		
=	TOTALS										
CC											

mask file PORTFOLIO

PORTFOLIO APPRAISAL

02/28/82

STUART DOMBER

SECURITY NAME	TICKER SYMBOL	COST	NO OF SHARES	UNIT COST	UNIT VALUE	MARKET VALUE	UNREALIZED GAIN--LOSS	% DIF	DIV/ INT.	ANNUAL INCOME
AMERICAN BAGLE	ABC	16,712.50	600	27.854	24.750	14,850.00	-1,862.50	-11.1	1.20	720.00
DELTA ECOLOGY FUND	DEF	497.00	200	2.485	25.625	5,125.00	4,628.00	931.2	2.20	440.00
CASH RESERVE MGT	CASH	1,250.00	1250	1.000	1.000	1,250.00	-	-		
GENERAL HOTEL IND.	GHI	17,600.00	600	29.333	25.125	15,075.00	-2,525.00	-14.3	1.60	960.00
JAMAICA KILO LTD	JKL	14,650.00	1200	12.208	27.000	32,400.00	17,750.00	121.2	2.68	3216.00
MAINE NOME ORCH.	MNO	13,562.50	500	27.125	41.750	20,875.00	7,312.50	53.9	0.93	465.00
PRINTING QUICK RE.	PQRC	6,875.00	500	13.750	16.375	8,187.50	1,312.50	19.1	0.16	80.00
STAR TAX UNITED	STU	12,712.50	300	42.375	19.750	5,925.00	-6,787.50	-53.4	0.44	132.00
VICTORY WATER EXCH	VWX	6,075.00	200	30.375	10.000	2,000.00	-4,075.00	-67.1	0.60	120.00
YELLOW ZONE FARMS	YZ	10,562.50	500	21.125	29.500	14,750.00	4,187.50	39.6	1.92	960.00
ALL AMERICAN AGRI.	AAAA	10,000.00	4000	2.500	2.125	8,500.00	-1,500.00	-15.0	0.10	400.00
ALFA ASTRO ARMS	AAA	8,000.00	1000	8.000	11.375	11,375.00	3,375.00	42.2	0.30	300.00
BETTER BUY BONDS	BBB	25,000.00	1000	25.000	32.813	32,813.00	7,813.00	31.3		
COUPON CAMERA CO	CCC	20,000.00	200	100.000	98.000	19,600.00	-400.00	-2.0	10.00	2000.00
TOTALS		163,497.00				192,725.50	29,228.50	17.9		9,793.00

KEYSTROKE MACRO:

PUTTING LISTS SIDE-BY-SIDE

Suppose that you have two lists in a file, one appearing just after the other, and you want to place them side-by-side. You can use scissors and paste, or the Merge function.

Here is a little macro that does the job, too:

First, some preparations: clear all tabs, and set a tab at the point List 2 should have its left margin after the relocation.

LIST NO. 1

1. Jones, R.
2. Ming, E.
3. Hemmingway, E.
4. Ageweeanon, K.
5. Rogers, G.
6. Rogers, R.
7. Rogers, J.
8. Eddie, N.

LIST NO. 2

1. Boler, S.
2. Simpson, G.
3. Hemmingway, M.
4. Chartoff, M.
5. Baez, J.
6. Allen, K.
7. Troy, H.
8. Philips, S.

Now, the macro:

1. Locate the cursor in col. 1 of the line containing List 2's first item.
2. <CTRL> B.
3. Clear the buffer.
4. Copy line into buffer (N.B., not "move" but "copy").
5. Up, one line (in other words, back to the line we began upon).
6. Line erase (N.B., not "delete" but "erase").
7. Cursor up, to item 2, List 1.
8. Dump buffer.
9. Up, one line.
10. Tab (see above).
11. Join lines.
12. Locate cursor in col. 1, of the line containing List 2's second item.
13. <CTRL> X -- IT'S DONE!

Now, touch <CTRL> X successively to relocate List 2's elements. It works on the List 2's title, too!

TYPING THE ENVELOPE

Here is an easy way to get the envelope for a business letter typed.

Prepare your letters in a consistent manner so that the first line below the address can always be indentified by searching for a particular string. Also, use a print command (eg. "^10") to space the letter down the page, and put the date well off to the right.

Create a permanent file called "envelope" on disk and put the following command in it:

```
1 DROP ^ FIND Dear KEEP xxxx 1 ARRANGE 100 135 1 50 PRINT 0 0
```

^10

John Doe
1234 Main Street
New York, N.Y. 10045

June 15, 1980

Dear Mr. Doe:

All work and no play makes Jack a dull boy. All work and no play makes Jack a dull boy. All work and no play makes Jack a dull boy. All work and no play makes Jack a dull boy. All work and no play makes Jack a dull boy. All work and no play makes Jack a dull boy. All work and no play makes Jack a dull boy.

All work and no play makes Jack a dull boy. All work and no play makes Jack a dull boy. All work and no play makes Jack a dull boy. All work and no play makes Jack a dull boy. All work and no play makes Jack a dull boy.

Sincerely,

Jack

After you have printed and saved your letter, give the command below:

```
INSERT ENVELOPE DO
```

You can fiddle with the column positions specified for the Arrange function to change the indentation of the address.

Answer To Walter E. Lewis

An abstraction of your problem, I think, is captured in the table below. The Tax is to be the Rate times the Net. You want to enter the data shown, and have the correct values put into the areas in braces. Everything else you want blank.

ex		Net		Rate		Tax
zv		999,999		999.99		999,999
+	Item A	10,000				
+	Item B	20,000				
= +	Total Class 1	{	}	10.00	{	}
+	Item C	5,000				
+	Item D	5,000				
= +	Total Class 2	{	}	5.00	{	}
= +	Grand Total				{	}
-	Exemption					100
=	Total Due				{	}

The first difficulty, as you noted, is that when you say "Compute" the rate you have entered disappears. This is because the rows on which you have put the rates are derived from other rows (namely, the sum of the two rows above). A row equation is always done for all the columns in a table, so the values you put are overwritten by the row equation. A way around this is to enter your rate by some other means and not as a data value on the row. Below, the rate figures are loaded into the correct columns by a column operation. A column operation or calculation is always done on a suitable row only after the row equations to derive it have been completed. Notice that "just compute" has been used so that just the next row is effected and that the number "1" has been assigned, since this is presumably the first thing to do on the row.

ex		Net		Rate		Tax
zv		999,999		999.99		999,999
+	Item A	10,000				
+	Item B	20,000				
jc1=				10.00		
= +	Total Class 1	{	}	{	}	}
+	Item C	5,000				
+	Item D	5,000				
jc1=				5.00		
= +	Total Class 2	{	}	{	}	}
= +	Grand Total				{	}
-	Exemption					100
=	Total Due				{	}

Now, the correct column equation has been added in the table below. "rc" is used so that it will only be applied to derived rows and the number "2" has been assigned so that it will be done in sequence after the constant operation.

Presumably, however, this equation should not be applied to any rows after the two subtotals, so it should be cancelled at that point. Also, you need a new example line to suppress printing of the meaningless row totals.

ex		Net 999,999	Rate 999.99	Tax 999,999
zv				
rc2		+	*	=
+	Item A	10,000		
+	Item B	20,000		
jc1=			10.00	
= +	Total Class 1 {	}	{	}
+	Item C	5,000		
+	Item D	5,000		
jc1=			5.00	
= +	Total Class 2 {	}	{	}
cc2				
ex		,,,,,,	,,,,,,	999,999
= +	Grand Total			{
-	Exemption			100
=	Total Due			{

Taking a tip from the pervious issue of this newsletter, the problems of cancelling column equation 2 and forcing blanks (or zeros) into the meaningless columns could be solved with one new column equation. In the table below, the number "2" is reassigned to a new equation which effectively cancels the old one with the same number. The new equation simply sets the first two columns to zero for all following rows. Remember that "fetching" a value that was never stored produces a zero.

ex		Net 999,999	Rate 999.99	Tax 999,999
zv				
rc2		+	*	=
+	Item A	10,000		
+	Item B	20,000		
jc1=			10.00	
= +	Total Class 1 {	}	{	}
+	Item C	5,000		
+	Item D	5,000		
jc1=			5.00	
= +	Total Class 2 {	}	{	}
ac2		ftz	ftz	
= +	Grand Total			{
-	Exemption			100
=	Total Due			{

OPENING AN IRA

The table below shows the value you will accumulate by opening an IRA account and putting \$2,000 per year into it. It also shows how much you can get out if you have to pull out early and pay both income tax and the withdrawal penalty of 10 percent. You can vary the amount of your contribution (a), the interest rate (b), and your percentage tax bracket (c). You don't have to keep the same assumptions for all years as is done here. If you want to change the interest rate in 1987, for example, just give a new "uc2" equation ahead of the row for 1987 with all the values to be used for following years.

Even if you have to withdraw and pay the penalty in five or so years, an IRA can still be a valid investment alternative if you are in a high tax bracket. You can compare the figures in the last column with other ways you might invest the money, but remember that if you invest it other ways, you will have to pay taxes on the money and can only invest what is left over.

	Year	Contribution	Interest Rate	Year End Value	Tax Bracket	Withdrawal Penalty	Withdrawal Amount
ex	9999	999,999	999.99	9,999,999	999.999	999.999	999,999.99
uc2=		2000a	10.00b		50.000c	10.000	
uc3		+	£	fta + £	+	+	s£h
uc4	pas	+		+ fta + = sta+			££
+	1982	2,000	10.00	2,200	50.000	10.000	880.00
uc1+	1						
+	1983	2,000	10.00	4,620	50.000	10.000	1,848.00
+	1984	2,000	10.00	7,282	50.000	10.000	2,912.80
+	1985	2,000	10.00	10,210	50.000	10.000	4,084.08
+	1986	2,000	10.00	13,431	50.000	10.000	5,372.49
+	1987	2,000	10.00	16,974	50.000	10.000	6,789.74
+	1988	2,000	10.00	20,872	50.000	10.000	8,348.71
+	1989	2,000	10.00	25,159	50.000	10.000	10,063.58
+	1990	2,000	10.00	29,875	50.000	10.000	11,949.94
+	1991	2,000	10.00	35,062	50.000	10.000	14,024.93
+	1992	2,000	10.00	40,769	50.000	10.000	16,307.43
+	1993	2,000	10.00	47,045	50.000	10.000	18,818.17
+	1994	2,000	10.00	53,950	50.000	10.000	21,579.99
+	1995	2,000	10.00	61,545	50.000	10.000	24,617.99
+	1996	2,000	10.00	69,899	50.000	10.000	27,959.78
+	1997	2,000	10.00	79,089	50.000	10.000	31,635.76
+	1998	2,000	10.00	89,198	50.000	10.000	35,679.34
+	1999	2,000	10.00	100,318	50.000	10.000	40,127.27
+	2000	2,000	10.00	112,550	50.000	10.000	45,020.00

Percents, Percents, and More Percents

=====						
Row Percents						
=====						
		----Store A----		----Store B----		
		Amount	Percent	Amount	Percent	
=====						
ex		999,999.99	999.99	999,999.99	999.99	,,,
ac1		sta	fta	stb	ftb	
ac2			+		+	=
ac3			+		+	pct
=====						
+	Payroll	132,678.80		142,184.10		
+	Benefits	21,567.91		23,488.12		
=====						
= +	Staff Costs					
=====						
+	Rent	62,500.00		62,500.00		
+	Utilities	18,145.61		20,156.40		
=====						
= +	Building Costs					
=====						
=	Total					
=====						

=====					
Column Percents					
=====					
		-----1981-----		-----1982-----	
		Amount	Percent	Amount	Percent
=====					
ex		999,999.99	999.99	999,999.99	999.99
uc1		+ fta/%rnd		+ ftb/%rnd	
+	Payroll	132,678.80		142,184.10	
+	Benefits	21,567.91		23,488.12	
=====					
= +	Staff Costs				
=====					
+	Rent	62,500.00		62,500.00	
+	Utilities	18,145.61		20,156.40	
=====					
= +	Building Costs				
=====					
jc1		sta		stb	
=	Total				
=====					

This table requires that Compute be used twice in succession (ie. "compute compute").

Percents, Percents, and More Percents

=====				
Row Of Percent Changes				
=====				

jcl	pch	+	+	+	+
=	Percent Change				
=====					

=====				
Percentage Change Column				
=====				
		Amount	% Change From Previous Year	
=====				
ex		999,999.99		999,999.99
zv				
uc2		sta		
+	1975	18,145.61		
uc1		*		fta + ~
+	1976	21,567.91		
+	1977	32,500.00		
+	1978	45,187.34		
+	1979	44,678.80		
=====				

Percents, Percents, and More Percents

One Value As The Percent Of Another

	Sales	Costs	Profit	Percent Profit
ex	999,999	999,999	999,999	9999.99
uc1	+	-	=	
ac2	/		+	%
+ +	Item A	123,714	100,856	22,858
+ +	Item B	67,145	51,907	15,238
+ +	Item C	47,119	33,102	14,017
= ///	Total			
ex	9999.99	9999.99	9999.99	, , , , , ,
%	Percent Item A			
%	Percent Item B			
%	Percent Item C			

Some Fixed Percents

	Cost A	Cost B	Overhead (10% of A + B)	Total
ex	999,999	999,999	999,999	999,999
zv				
uc1=			10	
uc2	+	+	* #	
uc3	+	+	+	=
+	Item 1	100	200	
+	Item 2	450	800	
= + +	Total			
cc				
*	Percent Tax	5	5	5
= #	Total With Tax			

