

# Spreadsheet

Spreadsheet (SPRED-sheet) programs calculate numbers. Whereas a handheld calculator displays one number at a time, a computer spreadsheet lets you *spread* many numbers over an entire *sheet* of your screen and label the numbers so you know what they refer to. The two most important features of spreadsheets are:

◆ **Formulas**

Instead of having to manually calculate numbers, you can create formulas that calculate for you.

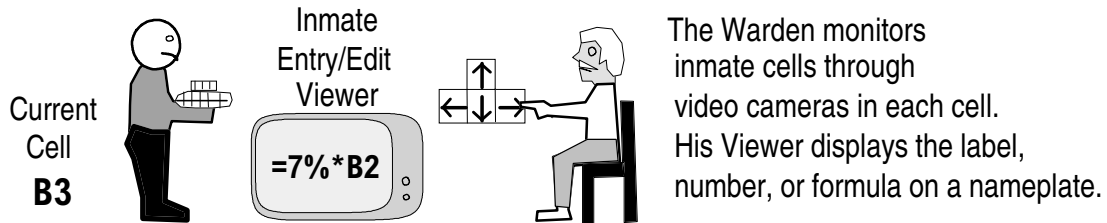
◆ **Automatic Recalculation**

When you change the numbers in your spreadsheet, the formulas automatically recalculate new answers. This makes it very easy to perform “what-if” scenarios. Examples: *What* would yearly profits be *if* we reduced costs by 7%? *What* would yearly profits be *if* we increased prices by 3%?



**Spreadsheet Jail Analogy**

A spreadsheet consists of vertical columns and horizontal rows that intersect and form *cells*. Imagine that a spreadsheet is a jail whose cells are identified by their column letters and row numbers. As new inmates arrive, they are given uniforms with identifying nameplates, then assigned to cells. Each cell can be viewed by the warden through hidden video cameras. If he sees a problem with a nameplate, he insists that it be fixed, either by sending it to the Entry/Edit Area or by changing it directly in the cell.



	A	B	C	D
1	The jail accepts three types of inmates.		<b>Labeled Leader:</b> He's a model inmate and has been put in charge of a related group of inmates. The label on his nameplate identifies the column (or row) he leads. He prefers to stand on the LEFT side of his cell.	
2			<b>Numbered NoName:</b> He arrived so recently, he got only an anonymously-numbered nameplate. He prefers to stand on the RIGHT side of his cell.	
3			<b>Formula Foreman:</b> He calculates the formula printed on his nameplate [=7%*B2] (which can be seen on the warden's video screen), then holds up a sign showing the numerical result of his calculation [ 7 ] (which hides the actual formula on his nameplate).	



Which cell contains the label ITEM 1? \_\_\_\_\_  
 What value is stored in cell B2? \_\_\_\_\_  
 What formula is stored in B3? \_\_\_\_\_ What value displays? \_\_\_\_\_  
 Answers: B1, 100, =7%\*B2, 7



*Since so many elements of a spreadsheet are interrelated, I've arranged these terms in functional, as opposed to alphabetical, order.*

In addition to those described on the following pages, spreadsheet programs typically have these features which were described in the Word Processing section:

- Alignment
- Macro
- Cut/Copy/Paste
- Margins
- Find & Replace
- Orientation
- Font
- Page Breaks

### SPREADSHEET USES

Spreadsheets were among the first integrated programs and combined these three applications:

- Spreadsheet—calculations, forecasting “what if” scenarios.
- Graphics—pie, bar, stacked, line graphs.
- Database—lists of data for fast search and retrieval.

Here are just a few of their many uses:

#### Accounting and Financial

Budgets, balance sheets, income statements, cash flow statements, billing, payroll, checkbook ledger balance, tax table lookup, amortization, depreciation, loan payments.

#### Sales/Marketing/Personnel

Automated price quotes, market surveys, questionnaire statistics, inventory lists, employee data, client data.

### WORKSHEET (SHEET)

A worksheet is one sheet of a spreadsheet file. It's composed of rows and columns that intersect to form cells. Early spreadsheet programs had only one worksheet. Related sets of data could be placed on different areas of the worksheet and linked together, but it was a cumbersome process.

	A	B	C	D
1				
2				
3				
4				

### WORKBOOK (NOTEBOOK, 3D SPREADSHEET)

A workbook is a collection of worksheets in a single spreadsheet file. Related data can be placed on successive worksheets and be easily linked.

	A	B	C	D
1				
2				
3				
4				

	A	B	C	D
1				
2				
3				
4				

	A	B	C	D
1				
2				
3				
4				

## FRAME (BORDER)

The frame is the top and left border of the worksheet. It's usually shaded and contains column letters and row numbers in boxes. The boxes for the current column and row typically appear pushed in or highlighted.

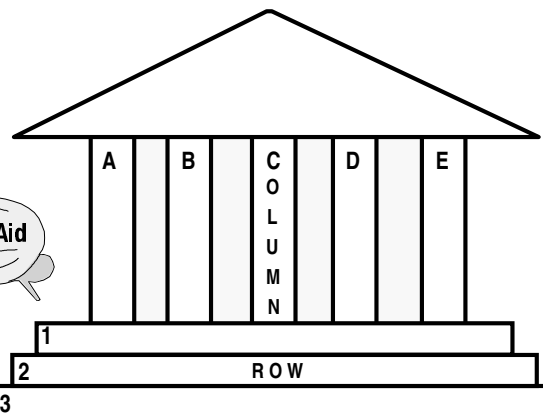
	A	B	C	D
1				
2				
3				

## ROW

A row is the horizontal component of a worksheet.

Analogy: a stair in front of a building.

Most spreadsheets have thousands of rows, in some multiple of 2, such as 4096, 8192, 16,384, 65,536.



## COLUMN

A column is the vertical component of a worksheet.

Analogy: a column holding up a building.

Most spreadsheets have 256 columns.

The first 26 columns are lettered A-Z followed by Section A lettered AA-AZ followed by Section B with BA-BZ, and so on. The 256th column is IV (Section I, column V). **BrainAid:** IV=Intravenous.

The math: 10 sections (A, AA, BA, CA, DA, EA, FA, GA, HA, IA) x 26 letters = 260 minus 4 (w, x, y, z) = 256.

	Section A					Section B					Sections C-H				Section I							
	A	B	C	....	Z	AA	AB	AC	....	AZ	BA	BB	BC	....	BZ	.....	IA	IB	IC	....	IV	
1																.....						
2																.....						
3																.....						
4																.....						

## CELL

A cell is a rectangle formed by the intersection of a column and row.

**Cell pointer:** a border or highlight that indicates the current cell. The cell pointer can be moved with the mouse or with cursor movement keys.



	A	B	C	D
1		ITEM 1		
2	PRICE	100		
3	TAX	=7%*B2		

## CELL CONTENTS

Cells can hold three things (LNF):

- Labels (text) to identify the numbers on the spreadsheet.
- Numbers (values) used for calculations or reporting.
- Formulas to calculate and display results (values).

Which cells have labels? \_\_\_\_\_

Which cell has a number? \_\_\_\_\_

Which cell has a formula? \_\_\_\_\_

Answers: Labels: A2, A3, B1; Number: B2; Formula: B3

## CELL ADDRESS

A cell address is the column letter and row number of a cell. For example, the address of the top left corner cell is A1. Some programs call cell addresses cell *references*, but I prefer to limit that term to cell addresses used in formulas. (See Formula).

## DATA ENTRY/EDIT

### Entering Data

Rule #1: Start In the Right Place and Proceed in the Right Order (SIRP-PRO).

**Trap!** Beginners forget to move the cell pointer to a new cell before entering new data and end up overwriting the existing data in the current cell.

Spreadsheet data entry is simple. Here are the typical steps:

1. Move the cell pointer to the desired cell.
2. Type in a label, number, or formula. (What you type appears in the Entry/Edit area and possibly in the cell as well. But the data isn't stored in the cell just yet.)
3. Tap [Enter] or a cursor movement key to store the data in the cell. (In general, tapping [Enter] after data entry keeps the cell pointer in the entry cell and its contents viewable in the Entry/Edit area. Tapping a movement key, however, deposits the data and moves the cell pointer.)

**Trap!** Some programs have an automatic-advance option which moves the cell pointer to the next cell when you tap [Enter]. The disadvantage is that you may not want to move in that direction after each entry. Unless you have a long data series to enter in one direction only, disable the automatic-advance feature.



If the inmate being admitted has a history of violence and the warden wants to keep an eye on him through the video viewer to make sure he doesn't damage his cell, tap [Enter]. If you have a nonviolent criminal, or have a long line of inmates waiting to check in, tap an [arrow] key to place the current inmate in his cell and simultaneously move to the next cell.

### Canceling Data Entry

If you are just starting to enter data into a cell and decide against it, or realize you're in the wrong cell, you can cancel the entry, typically by tapping the [Escape] or similar key.

**BrainAid:** Pretend your prisoner escaped while you were checking him in.

### Replacing Data

New data you type in a cell automatically replaces existing data. You do NOT need to erase existing data.

**BrainAid:** Pretend you are replacing the inmate's nameplate with a totally new one.

### Editing Data

To make a small change to existing data, it's more efficient to edit it than to replace it. Usually you press a certain key or use mouseclicks to ready the data for editing, either in the cell itself, or in the Entry/Edit area.

**BrainAid:** Rather than create a new nameplate, you fix the incorrect character on the existing one.

### Erasing Data

To erase data, go to the desired cell and tap the [Delete] key. **BrainAid:** Delete discharges an inmate.

**Trap!** You can NOT erase with the [Spacebar] key. Cell contents may appear to be deleted, but they are actually replaced with a space character (ASCII 32).

**BrainAid:** Tapping [Spacebar] is equivalent to putting an invisible inmate into the cell. Since invisible inmates are impossible to monitor, the warden insists that you [Delete] them.

## MODE

The mode is an indicator of the current spreadsheet condition. For example, many spreadsheets indicate that they are in *Ready* mode when they are ready receive data or a command from you. They switch to *Entry* mode (or perhaps *Label* or *Value* mode) while you are entering data and to *Edit* mode while you are editing data. Other modes may include *Point*, *Error*, *Wait*, *File*, and *Menu*.

## RANGE (BLOCK)

A range is a single cell or a group of cells. Similar to a block in word processing, ranges let you select a portion of a spreadsheet and perform operations on it. You can change font, format, move, copy, delete, shade, border, etc. You can also specify a range in a formula, for example, to add the contents of all cells in that range. Typically, ranges must be rectangular, but some spreadsheet programs let you select multiple, nonadjacent ranges.



	A	B	C	D
1	A1		C1:D1+	
2				D2
3	A3:B4			
4	A3:B4			

Is A1 a range? \_\_\_\_\_

Is C1:D1+D2 a range? \_\_\_\_\_

Is A3:B4 a range? \_\_\_\_\_

Answers: Yes to all, but technically C1:D1+D2 is two adjacent ranges.

### Range Addresses

A range address is designated by a start cell and an end cell typically separated by a colon or by two periods.

Examples:

- Single Cell—A1
- Partial Row—A1:D1 or A1..D1 (observe that all cells are in Row 1).
- Partial Column—A1:A4 or A1..A4 (observe that all cells are in Column A).
- Rectangle—A1:D4 or A1..D4 (observe that this range covers 4 columns and 4 rows).

### Range Names

A range name is a word used to identify a range of cells. Range names are easier to work with than obscure cell addresses. For example, if you name the range HD2975:IG3357 as SALES, it's much easier to create and understand the function SUM(SALES) than it is to deal with the unwieldy SUM(HD2975:IG3357).

## LABEL / LONG LABEL

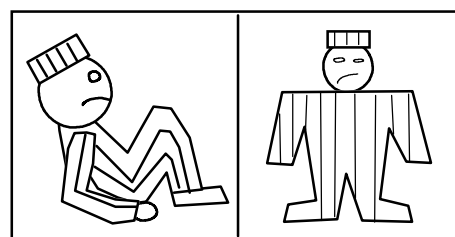
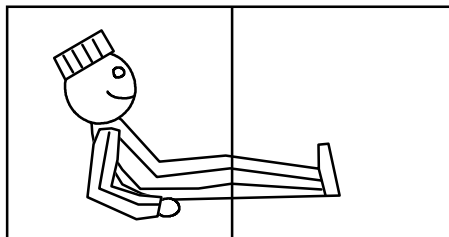
A label is a word or words used to identify numbers on a worksheet. A *long* label exceeds its cell width and extends into the *unoccupied* cell(s) to the right of it. Although it appears to be in multiple cells, a long label resides **ONLY** in its start cell. If you enter data into the cell on the right, it appears to truncate, that is, cut off, the long label. However, the label can be viewed in its entirety in the Entry/Edit area.

**Trap!** You unknowingly have an invisible space character in the righthand cell which appears to truncate the long label. You'll need to delete the space character to see the long label.

The typical spreadsheet cell default width is about 8 or 9 characters. If you have a label that exceeds this length you can always widen the cell, but this also widens the entire column, which might not be appropriate if you intend to enter other data into the column. So if you need, for example, a title to go across the top of several columns, it's best to use the long label feature.



Imagine that a long label is like a tall inmate who stretches his legs into the empty cell to his right. This works fine until a new inmate is admitted. Then the tall inmate must pull his knees up and scrunch his long legs back into his own cell. His legs appear to be cut off, but the warden can always see them in the video viewer. Not surprisingly, the tall inmate lives for the day that his neighbor is discharged. Only then will he be able to stretch out his legs again!



## NUMBER / LONG NUMBER

A number is a value used in a worksheet for calculations or reporting. A *long* number is one that is too wide to display in its cell. Rather than truncate the end of a long number (like is done with a long label) and risk your assuming it's accurate, most spreadsheets display a string of warning symbols like ##### or \*\*\*\*\*.

**Note:** A spreadsheet may automatically display very large or very small numbers in scientific notation; for example, 570,000,000,000 may be displayed as 5.7E+11 (5.7 times 10 to the eleventh power), or 0.0000000000095 may be displayed as 9.5E-12 (9.5 times 10 to the -12 power).

**Tip:** To display long numbers try the following:

- Increase the zoom (aka magnification) setting.
- Widen the cell (which actually widens the entire column).
- Choose a smaller format (to reduce extra symbols in a number, for example, \$5,000.00 vs. \$5000).



Imagine a numbered inmate slips in his cell, hits his head on the bars, and see stars \*\*\*\*\* or curses ##### as his head swells beyond the cell width.

## SERIAL NUMBER - DATE & TIME

A serial number is a number used to represent a date and/or a time. Serial numbers make date and time arithmetic possible. To most spreadsheets, history started on January 1, 1900 which was assigned serial number 1. Each day after was given the next whole serial number. Date examples:

2 = January 2, 1900 (the second day of the century)

31 = January 31, 1900 (the number of days in the first month)

366 = December 31, 1900 (the number of days in the first year—must have been a leap year!)

36525 = December 31, 1999 (the number of days in the first 100 years—including 25 leap years)

73050 = December 31, 2099 (the ending date varies by spreadsheet brand).

If your birthday were October 7, 1952, for example, typing a Date function (syntaxes vary) like DATE(1952,10,7) into a cell displays the serial number 19274, indicating your day in the century.

To figure out the number of days between, for example, April 27, 1998 and June 16, 1999, you could laboriously count the days on a calendar, or you could use Date functions to figure it out for you with a formula like DATE(1999,6,16)-DATE(1998,4,27) which returns 315 days.

Time is treated as the decimal part of a serial number based on a 24-hour clock. Time examples:

.00000 = 0000 hours = 12 am (midnight or 2400 hours)

.33333 = 0800 hours = 8 am (1/3 of a 24-hour day)

.50000 = 1200 hours = 12 pm (noon: 1/2 of a 24-hour day)

.66666 = 1600 hours = 4 pm (2/3 of a 24-hour day)

.99999 = 2359 hours = 11:59 pm (the last minute of the day)

**Tip:** To convert p.m. time to 24-hour time, add 12:00; for example 5:23 pm + 12:00 = 1723 hours. To convert back, deduct 1200; for example, 1946 - 1200 = 7:46 pm. (Note: a.m. time matches 24-hour time.)

To perform time arithmetic, say to bill a client for consulting services from 9:32 am to 2:57 pm, you could create a formula like TIME(14,57,0)-TIME(9,32,0) which returns the fractional part of the day as .225694, then multiply that fraction by 24 to find the number of hours is 5.42.

**Tip:** If you type NOW into a cell (syntaxes vary) it will return the current serial [whole+decimal] number based on your computer's date/time setting. Each time you print, this cell will display the current date and/or time.

### Date/Time Formats

In most cases, you'll want serial numbers to remain behind the scenes and have their corresponding dates or times appear in the cell. To accomplish this, you must format each cell accordingly (see Format).

## FORMULA

A formula is a mathematical expression (phrase or sentence) that calculates and displays results. It can contain Cell References, Operators, and Numbers. **BrainAid:** CRON.

### Formula Syntax (Punctuation)

Typically you must start a formula with an equal sign which tells the cell you want it to "be equal to" the results of the formula you are typing.



**Trap!** If you forget the equal sign (or a + or –), the cell will treat your formula as a label which will not calculate.

**Trap!** Don't use a space character unless the formula contains a text string (which must be in quotes).

	A	B	C	D
1	9	5		
2	6			
3	=A1+A2			
4			Total	=A3

### Cell Reference

A cell reference is a Cell Address Used In Formula.

**BrainAid:** CAUIF. Rather than using actual values in a formula, cell references "refer" to other value-containing cells. Cell references are also called *precedents* [prih-SEE-dunts] as their values "precede" or come before the formula. Changing a precedent's value causes the *dependent* formula cell to recalculate.

Example: In the worksheet shown above, the calculated value in A3 = 9+6 = 15. If you change A2 to 10, then A3 = 9+10 = 19.

**Tip:** A cell reference placed in a cell by itself mirrors the contents of the referenced cell. In the worksheet shown above, D4 contains the reference =A3. Therefore, whatever appears in A3 will also appear in D4.

Is A1 a cell reference? \_\_\_\_\_

Is B1 a cell reference? \_\_\_\_\_

What will display in A3? \_\_\_\_\_

What will display in D4? \_\_\_\_\_

Is A3 a cell reference? \_\_\_\_\_

If A2=10, then A3= ? \_\_\_\_\_

Answers: Yes; No; 15; 15; Yes; 19.

### Priority Of Operators (PEMDAS)

When a formula has more than one mathematical operator, the spreadsheet performs them in standard math order following the acronym PEMDAS. In the examples below, I've bolded the operation that has priority.

- 1st: Parentheses—The operator inside has priority. **(3+4)\*10 = 7\*10 = 70.**
- 2nd: Exponentiation—Raising a number to a power has priority. **4\* 2^3 = 4\*8 = 32.**
- 3rd/4th: Multiplication and Division—Have left to right priority. **4\*5/10 = 20/10 = 2.**
- 5th/6th: Addition and Subtraction—Have left to right priority. **10-3+2 = 7+2 = 9.**

**Trap!** Ignoring PEMDAS priorities leads to incorrect results.

**Tip:** When building a formula with multiple operators, test it with several numbers (positive, negative, decimals, etc.) and verify the results.

### Point & Build

Point & Build is a feature that lets you "point" to cell references instead of typing them in. When typing in cell references character by character, it's easy to make a mistake; for example, you might accidentally transpose AV374 into AV347. A much more accurate and intuitive way is to first type the symbol that begins a formula (usually an = or a + sign), then use your mouse to point and click on the desired cell reference which will automatically appear in the formula cell. (You can also use the arrow keys to move and "point" to the desired cell.) Type the next operator and point and click on the next cell reference and so on until your formula is built. To enter a range of cell references, drag from the beginning cell to the ending cell.

### PEMDAS Matters!

To calculate the average of 4 and 6:

<u>Incorrect</u>	<u>Correct</u>
4 + <b>6</b> / 2	<b>(4+6)</b> / 2
4 + 3	10 / 2
7	5

### ToDo

Follow PEMDAS priorities to calculate the following:

$$5 + 2 * 3 = \underline{\hspace{2cm}}$$

$$(5+2) * 3 = \underline{\hspace{2cm}}$$

Answers: 11, 21

## FUNCTION\*

A function is a built-in formula. Spreadsheets typically have hundreds of functions that make creating formulas quicker and easier. Typical function categories include Date & Time, Financial, Logical, Lookup, Mathematical, Statistical, and Text.

One of the most commonly used functions is SUM, which adds a series of numbers. For example, if you wanted to add the contents of every cell from A1 to A15 you could create this long, tedious formula: (A1+A2+A3+A4+A5+A6+A7+A8+A9+A10+A11+A12+A13+A14+A15). But it would save a lot of typing and be less error prone to use a SUM function to add the cell range as in: SUM(A1:A15).

**Trap!** If you insert or delete rows above a SUM function, check to be sure the SUM range expanded or contracted accordingly.

Another particularly useful function is IF, which allows you to test for a certain condition and then respond in one of two ways like this:

IF(condition is true, do this, else do this)

For example, if you had a spreadsheet that calculated profit or loss, you could create a function that tested to see if Sales minus Costs were positive or negative, then display a Profit or Loss label accordingly.



	A	B
1	Sales	100
2	Costs	40
3	=IF(B3>0,"Profit","Loss")	=B1-B2

The IF function in cell A3 displays the label 'Profit' if the formula in B3 yields a positive value, or 'Loss' if the B3 value is negative.

What will display in A3? \_\_\_\_\_

Answer: Profit (since B3=60 and 60>0)

## ADJUSTMENT\*

Adjustment is a feature that updates copied formulas. When you copy a formula, its cell references 'adjust' to their new column/row location. This allows a formula pattern to be created once and then copied as needed, saving lots of time. Let's say your goal is to create formulas to total each of 12 columns. Without adjustment you'd have to type in 12 separate formulas, one per column. With adjustment you need type only the first column's formula, then copy it in one step to the 11 remaining columns.

Example: In the diagram shown here, A3 contains the formula =A1+A2 which adds the 2 cells above A3. If A3 is copied to B3, rather than making an exact copy, the formula adjusts to =B1+B2 which adds the 2 cells above B3. The same holds true for the formulas copied to C3 and D3—each formula's cell references have adjusted relative to its column.



When you ask the Formula Foreman in A3 (see diagram) what his job is, he replies, 'My job is to add the contents of the two cells above me.' You then have A3 train a new foreman in B3 to do the same job for column B. When you ask the B3 foreman what his job is he replies, 'My job is to add the contents of the two cells above me.' Conclusion: Relative to their respective locations, the foremen in A3 & B3 have the same job—which is of course, what you want.



	A	B	C	D
1	5	5	6	7
2	3	4	5	2
3	=A1+A2	=B1+B2	=C1+C2	=D1+D2

Copying the formula in A3 to the range B3:D3 results in three adjusted formulas, each doing the same task for their respective columns.

If you asked each foreman what his job was, he'd reply: 'My job is to:

Answer: add the two cells above me."

What is the advantage of adjustment?

Answer: Saves time with repetitive formulas.



## ABSOLUTE/RELATIVE/MIXED ADDRESSES

Absolute, relative, and mixed are the three types of cell addresses. **BrainAid:** ARM.

Inserting a dollar sign before a column letter or row number prevents it from adjusting when copied.

- Absolute Address [\$A\$1]  
Does NOT adjust when copied. Use this to hold a cell reference constant.
- Relative Address [A1]  
Adjusts to its new location when copied. This is the normal address type.
- Mixed Address [A\$1 or \$A1]  
Only the column or row adjusts when copied. Use this to hold a row or column reference constant.

Example: The tax formula in cell B3 has an absolute reference \$A\$1 (sales tax rate of 7.5%) and a relative reference B2 (price of ITEM 1). When you copy B3 to C3:D3, the price cells (C2, D2) adjust for each item, but the 7.5% tax rate (\$A\$1) remains fixed.

	A	B	C	D
1	7.5%	ITEM 1	ITEM 2	ITEM 3
2	PRICE	100	200	300
3	TAX	=\$A\$1*B2	=\$A\$1*C2	=\$A\$1*D2
4	TOTAL	=B2+B3	=C2+C3	=D2+D3

Instead of \$A\$1, you could have initially entered the 7.5% tax rate in B3 ( $=7.5\%*B2$ ) and copied B3 to C3:D3 ( $=7.5\%*C2$ ;  $=7.5\%*D2$ ), but the advantage of using an absolute reference is if the tax rate changes, you need only update it in A1 for *all* tax formulas.

## CIRCULAR REFERENCE

A circular reference occurs when you create a formula that contains its own address, or refers to another cell that in turn refers back to the formula cell. Circular references are usually unintentional and produce erroneous results, although some problems require a circular reference to produce the desired result.

Usually some type of indicator, such as CIRC or Circular, appears to warn you of this condition, but it won't prevent you from creating a circular reference. That's why it's always important to test your formulas with several sample values to make sure they calculate properly.

In the diagram shown, the formula in A3 contains a circular reference: the SUM range A1: A3 is equal to  $A1+A2+A3$ . But a formula to add the contents of A1 and A2 should not include A3. As written, the formula would calculate correctly the first time as  $9+6+0=15$ , and you might think everything is okay.

	A	B
1	9	
2	6	
3	=SUM(A1:A3)	

CIRC: A3

The circular reference cell, A3, will add its own value to the formula on each recalculation.

But on the next recalculation (which usually occurs each time new data is added to *any* cell), A3 already has 15 in it, and the formula would recalculate as  $9+6+15=30$ . On successive recalculations, the total in A3 would become  $9+6+30=45$ , then  $9+6+45=60$ , and so on. This would continue until you eliminate the circular reference by changing the formula to  $SUM(A1:A2)$ .

In some cases, a circular reference is created on purpose. Problem: A firm decides that employee bonuses are to be 20% of the firm's Net profit. But Net profit can't be calculated until the bonus is known since  $Net\ profit = Gross\ profit - Bonus$  (which hasn't been calculated yet!).

Solution: Since the Net depends on the Bonus which depends on the Net, create a circular reference to perform multiple iterations (recalculations) until a true result is achieved.

	A	B
1	Gross Profit	10,000
2	Bonus	20%*B3
3	Net Profit	B1-B2

CIRC: B3

Each iteration brings the values closer to the correct result.

## FORMAT

Format is a feature that changes the appearance but not the value of cell contents. In some spreadsheets, numbers must be entered without punctuation, so format commands are needed to alter their appearance. In all spreadsheets, format commands save time when changing the appearance of an entire range of numbers. Most spreadsheets also permit cell contents to be formatted with alignment and font attributes and cells to have borders and shading. Here are some numerical format examples:

<u>Raw Number</u>	<u>Currency Format</u>	<u>Percent Format</u>	<u>Scientific Format</u>
1357	\$1,357.00	135700%	1.36E+03

**BrainAid:** To impress visiting dignitaries, the warden issues dress uniforms to Numbered NoNames to wear over their plain, drab uniforms. But these fancy uniforms are façades (false fronts)—the plain, drab, raw number uniforms still appear in the warden's video viewer.

**ToDo:** Compare the meaning of the term format in an OS and in a word processor.

## FREEZE (TITLES)\*

Freeze is a feature that prevents row and/or column labels from scrolling off the screen. Otherwise, you'd be looking at a screenful of unlabeled numbers. You can horizontally freeze the top row(s) or vertically freeze the leftmost column(s), or freeze both. **Note:** Frozen titles *do* move with their respective rows or columns, remaining stationary only when scrolled *against*; that is, row labels freeze on up/down scrolls, column labels freeze on left/right scrolls.

	A	B	C
1		JAN	FEB
2	Sales	100	200
3	Costs	40	70
4	Profits	60	130

Original Spreadsheet

	A	B	C
2	Sales	100	200
3	Costs	40	70
4	Profits	60	130
5			

**No Freeze:** When you scroll down, Row 1 moves up, and you can't see the months.

	A	B	C
1		JAN	FEB
3	Costs	40	70
4	Profits	60	130
5			

**Freeze:** When you scroll down, Row 1 remains in place, so you always see the months.

## PROTECTION (LOCKED)

Protection is a feature that prevents changes or erasures to cell contents. Typically, you'll protect unchanging labels and formulas, but leave cells that may need updating unprotected.



Imagine covering the entire spreadsheet with a see-through layer of plastic, then cutting out portions of plastic to uncover the cells that may need changing.

	A	B	C
1		JAN	FEB
2	Sales	100	200
3	Costs	40	70
4	Profits	=B2-B3	=C2-C3

All cells are unprotected and can be changed or erased as desired.

	A	B	C
1		JAN	FEB
2	Sales	100	200
3	Costs	40	70
4	Profits	=B2-B3	=C2-C3

All cells are covered and protected. No changes or erasures are allowed.

	A	B	C
1		JAN	FEB
2	Sales	100	200
3	Costs	40	70
4	Profits	=B2-B3	=C2-C3

Certain cells are uncovered so data can be edited. Labels and formulas remain covered for protection.

## SCENARIO\*

A scenario is a named set of input values that can be swapped in and out of a worksheet model to test for different outcomes. Storing several values saves time over having to individually reinput each value.

## SPLIT (WINDOWS)\*

Split is a feature that divides a spreadsheet into two or more windows or panes. What appears to be two or more separate spreadsheets is actually the one spreadsheet seen from different viewpoints.

You can scroll each pane independently for viewing distant cells on the same screen. Or you can apply different formats to the same set of cells; for example, general format in one pane and currency format in another (see Format).

Example: Most 12-month spreadsheets are too wide to be seen at once across a typical computer screen.

	A	B	C	D	E	F	G	H
1		JAN	FEB	MAR	APR	MAY	JUN	JUL
2	Sales	100	200	300	200	400	200	100
3	Costs	40	70	140	30	90	70	40
4	Profits	60	130	160	170	310	130	60

But if the spreadsheet is split, the left pane can display January while the right pane displays December.

	A	B	C	D	J	K	L	M
1		JAN	FEB	MAR	SEP	OCT	NOV	DEC
2	Sales	100	200	300	500	400	600	200
3	Costs	40	70	140	100	190	300	90
4	Profits	60	130	160	400	210	300	110

## TEMPLATE\*

A template is a prebuilt worksheet with labels and formulas already entered, tested, and protected. You then fill in the desired data, and the results are automatically calculated.

Example: Checkbook Template

- Contains all labels and formulas needed in a checkbook register.
- You fill in the amounts for your checks, deposits, bank charges, etc.
- The template automatically calculates the ending balance for you.

**Trap!** If you build your own template, be sure to save it to a master file and make that file “read-only” so it can’t be altered (per your Operating System manual). Then, when you open the template, immediately resave it to a working filename before entering new data.

## UNDO\*

Undo is a feature that reverses the last editing/formatting action, returning the spreadsheet to its previous Ready state. A copy of the entire previous spreadsheet is kept in a RAM buffer for this purpose.

**Trap!** Compared to a word processing program with a multiple or unlimited undo feature, some spreadsheet programs undo only the last or last few actions. In some spreadsheets, saving empties the undo buffer so you can’t undo actions made before the save.

## SS SUMMARY



Look back through these pages and write down a brief description that will help you remember each item listed below.

Worksheet (Sheet)—

Workbook (Notebook, 3D Spreadsheet)—

Frame (Border)—

Row—

Column—

Cell—

Cell Contents—

Cell Address—

Data Entry/Edit—

Mode—

Range (Block)—

Label / Long Label—

Number / Long Number—

Serial Number - Date & Time—

Formula—

Function\*—

Adjustment\*—

Absolute/Relative/Mixed Addresses—

Circular Reference—

Format—

Freeze (Titles)\*—

Protection (Locked)—

Scenario\*—

Split (Windows)\*—

Template\*—

Undo\*—

# SS TermMap



Practice reciting the following terms in order from memory. Then briefly define each.

**Terms (16):** Worksheet, Row, Column, Range, Cell, Address, Reference, Label, Number, Formula, Function, Adjustment, Format, Freeze, Protect, Split.

**Acronym:** WRCR-CAR-LNF-FAF-FPS

**Acrostic:** Worker car linf faf fips.

	A	B	C
1			
2			
3			

**WORKSHEET**

	A	B	C
1			
2	Row		
3			

**ROW**

	A	B	C
1			
2		Column	
3			

**COLUMN**

	A	B	C
1			
2		Range	
3			

**RANGE**

	A	B	C
1			
2		Cell	
3			

**CELL**

**B2**

**ADDRESS**

	A	B	C
1			
2			
3			=B2

**REFERENCE**

	A	B
1		
2	Price	
3		

**LABEL**

	A	B	C
1			
2		125	
3			

**NUMBER**

	A	B	C
1			
2			
3			=3*B2

**FORMULA**

**SUM(A1:A15)**  
**FUNCTION**

Copy  
 $=A1+A2$      $=B1+B2$   
**ADJUSTMENT**

**5 - \$500**  
**FORMAT**

	A	B	C
1		Jan	Feb
2	Sales	35	47
3	Costs	52	30

**FREEZE**

	A	B	C
1		Jan	Feb
2	Sales	12	15
3	Costs	10	13

**PROTECT**

Split

	A	B	M
1		Jan	Dec
2	Sales	35	72
3	Costs	52	28

**SPLIT**

# DATABASE

A database is a "base" for storing data items. Just as military personnel and equipment are stored on a military base, data words and numbers are stored on a database. The two main database features are:

## ◆ Sorting

Scenario: Your boss plans to send holiday cards to the firm's employees. He shows you a filing cabinet containing over 10,000 alphabetically-sorted employee records. Being cost conscious, he wants you to sort them into zipcode order to qualify for inexpensive bulk-rate mailing, then type up envelopes to match. Of course, then he wants you to re-sort and re-file them (groan!) alphabetically. If you had to do this manually, the holiday might be over before you mailed the cards. But, if the records were stored in an electronic database instead of on paper, you could do this job in a day. You'd simply tell the computer to sort by zipcode, and it would be done in seconds. Then you'd tell it to print envelopes while you work on other tasks. Finally, you'd tell the computer to re-sort alphabetically and you're finished.

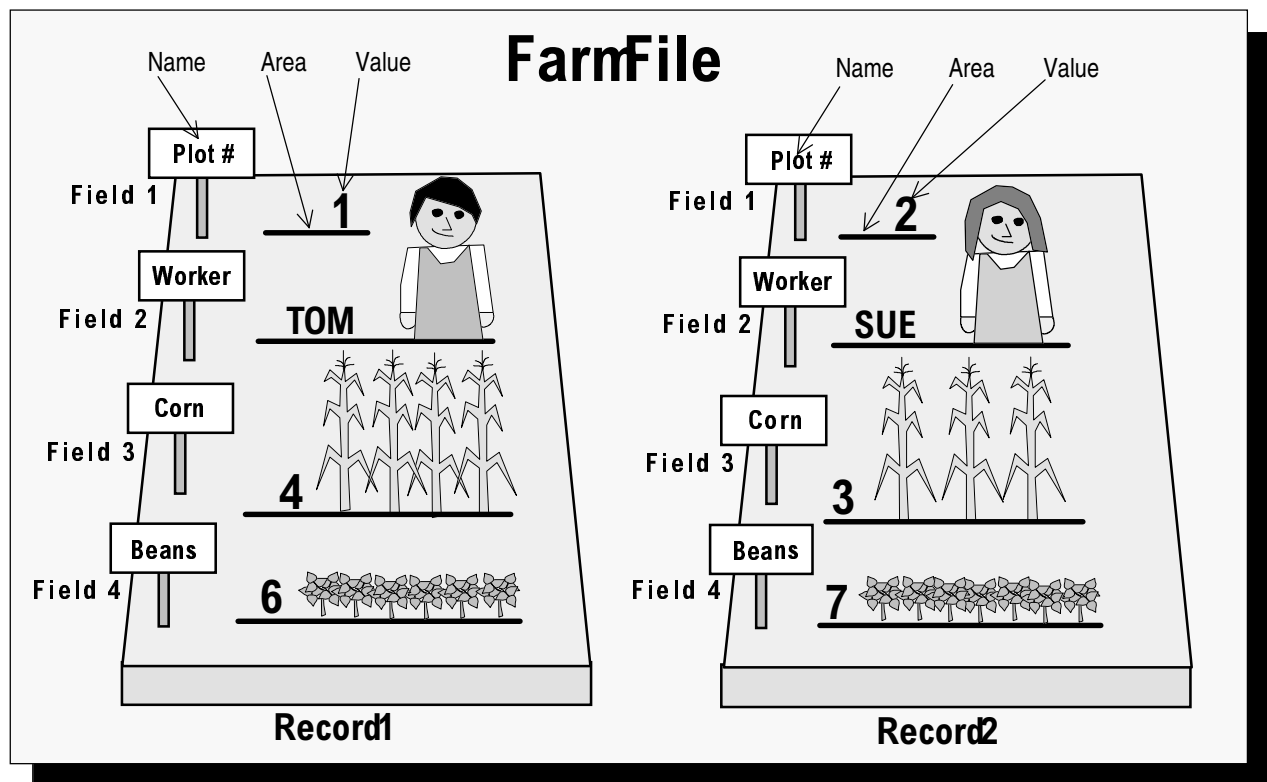
## ◆ Searching

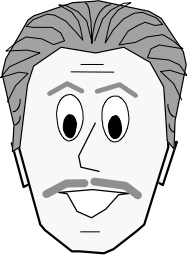
Scenario: A customer calls with a question about her bill. Out of millions of customers, you must find her particular record. If you had to physically search for a paper record, she might hang up in frustration before you found it. But, if the records were stored on an electronic database, you could find it in seconds! Electronic databases can even search for records based on multiple criteria, such as all married customers in zipcode 90210, who have no children, earn over \$250,000/year, and lease their cars. Don't try this manually!

## Down On The Database Farm



A database *file* contains *records* which contain *fields*. Let's compare a file to a farm, records to plots on the farm, and fields to pieces of the plots. Our farm has 2 plots. Each plot has 4 fields. Each field has 3 components: a name (on a sign) identifying the area (row in the dirt) into which values (seeds) are planted. In the first plot, we've "planted" a '1', Tom, 4 acres of corn, and 6 acres of beans. In the second plot, we've "planted" a '2', Sue, 3 acres of corn, and 7 acres of beans.





*Like a spreadsheet, many elements of a database are interrelated, so I've arranged the following database terms in functional, as opposed to alphabetical, order.*

**DATABASE USES**

A database can store any type of data: words, numbers, symbols, graphics, video, sound.  
 Examples: employee records, client records, inventory records, library catalog, hobby collection, phone/address lists, event schedules, sound clips, video clips, employee photos.

**FILE**

A database file is a collection of related records.  
**BrainAid:** Our farm *file* consists of a *pile* of paper records.

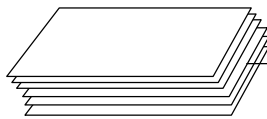
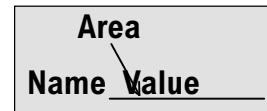
**RECORD**

A database record (REK-urd) is a collection of fields that describe an entity.  
 Entity = Person, Item, Place, or Event (PIPE). Examples: clients, inventory, cities, concerts.  
**BrainAid:** We keep a record for each farm plot on one *page* of paper.

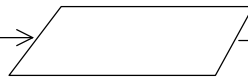
**FIELD**

A database field (feeld) is a named area of a record that holds a value.  
 A field has three parts: Name-identifies field. Area-holds data. Value-data (NAV).  
**BrainAid:** Our plot pages are divided into *pieces* called *fields*.

**Trap!** Don't confuse the field name (which is the same for all records) with the field value (which is usually different for each record). Field names that include the word "name" can be especially misleading. For example, in the case of [Name Bill] *Name* is the field name and *Bill* is the field value.



**File=File**



**Record=Page**



**Field=Piece**



Answer these questions about the Farm File.

How many records are shown? \_\_\_\_\_  
 How many fields per record are shown? \_\_\_\_\_

What is the entity being described? \_\_\_\_\_  
 What is the name of Plot #1' s second field? \_\_\_\_\_  
 What is the name of Plot #2' s second field? \_\_\_\_\_  
 What field part do the underlines represent? \_\_\_\_\_  
 What value is stored in Plot #1' s second field? \_\_\_\_\_; Plot #2' s second field? \_\_\_\_\_

<b>Plot#</b>	<u>1</u>
<b>Worker</b>	<u>TOM</u>
<b>Corn</b>	<u>4</u>
<b>Beans</b>	<u>6</u>

<b>Plot#</b>	<u>2</u>
<b>Worker</b>	<u>SUE</u>
<b>Corn</b>	<u>3</u>
<b>Beans</b>	<u>7</u>

Answers: 2; 4; Plot; Worker; Worker; Area; TOM; SUE.



## KEY\*

A key is a field used for searching or sorting. It can be *any* field you choose.

### Search Keys

While you can search for a database record using *any* field as a search key, it's best to choose a field that contains no duplicate values. When your key field has a different value for each record, the computer is sure to find the record you want.

#### Poor Search Keys

Lastname—You're likely to have several people with common names (for example, Smith) in a database.

Model Number—You're likely to have items (for example, computers) with the same model number.

#### Good Search Keys

Social Security Number—No two are alike.

Serial Number—No two are alike.

#### Multiple-Key Search

To find records that meet multiple criteria, you'd search on multiple field(s) (see Query).

Example:

Married = Yes

AND

Zipcode = 92010

OR

Zipcode = 92012

AND

Credit = Good

### Sort Keys

You can sort on *any* field desired, but if duplicates exist, you'll need to sort on multiple fields. To get the best results, think out your sort order beforehand. In most cases you'll want to sort first on a general category, then on a subcategory within the larger one, and so on.

Example:

1st or Primary Key: Zipcode (to sort records into geographical zones).

2nd or Secondary Key: Lastname (to sort records by lastname within each zipcode).

3rd or Tertiary Key: Firstname (to sort records by firstname for each common lastname, like Smith).



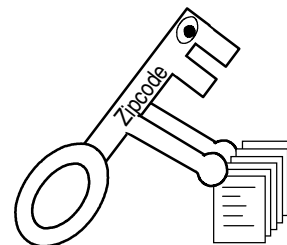
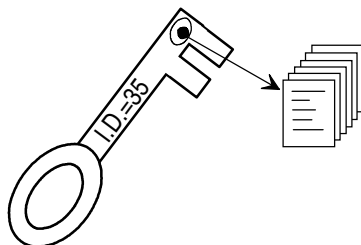
A key field contains the "key" to unlock desired values.

#### Searching

A "key" with an eye, searching for a record.

#### Sorting

A "key" with hands, sorting records.



## DBMS

A DBMS (dee-bee-em-ess) is a set of programs that can create, modify, and manipulate database data. DBMS = Database Management System. A database itself is merely a collection of data. To work with data, your computer needs DBMS program instructions to tell it what to do.

### DBMS Features (PRESS)

Here are just some of the things a DBMS can do:

- Prints reports, labels, or envelopes.
- Retrieves records rapidly.
- Extracts records that meet specified criteria.
- Sorts alphabetically, numerically, by date, etc.
- Saves paper and storage space.

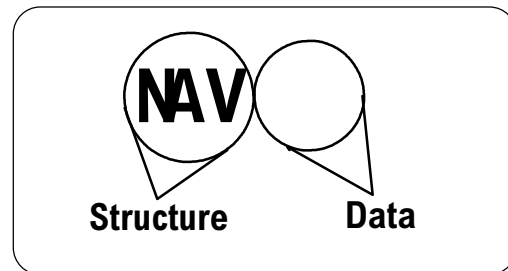
### DBMS Architecture

The two primary elements of a DBMS are:

I. Structure

II. Data

First you design and build the Structure, which consists of field Names and Areas; then you enter Data, in the form of field Values, into the Structure.



### DBMS Classifications

A DBMS can be classified as:

- RAM-based or Disk-based
- Flat-File or Relational

A spreadsheet program, when used to store data, is RAM-based and Flat-File. A simple database program, known as a List-Manager, is Disk-based and Flat-File. A commercial database program is Disk-based and Relational.

**RAM-based vs. Disk-based**

**Flat-File vs. Relational**

I' ll expand on these classifications later.



*Max, I' ve heard that a database application is harder to learn than a word processing or spreadsheet application. Is that true?*

Yes and no, K.N. It's not that any one thing about a DBMS is hard to learn, it's just that there are so many things, all with their own screens and commands for manipulating data, that it's easy to get confused.

The best thing you can do in preparation for learning a database application is to be sure you are very clear on database terminology, especially the differences between:

Record vs. Field

Field Name vs. Field Value

Structure Table vs. Data Table

## STRUCTURE (DESIGN)

Structure is a database's framework of field names, areas, types, and properties. The toughest part of designing database structure is figuring out which fields to include—no matter how hard you try, it seems you always overlook a needed field. It's possible to change field definitions *after* you've entered data, but you might lose data or have to enter data you left out. It pays to design carefully from the start.

### Field Name

Farm analogy: Paint a field's name on a sign to identify what is to be planted in that area. Some DBMSs allow spaces in field names, others don't. Most DBMSs have a maximum limit for the characters in a field name.

### Field Area (Width, Size)

Farm analogy: Plow a row. **Trap!** If you reserve too little field area, your longer data items will be truncated (cut off). But if you reserve too much area, you'll waste disk space (a million records with just one unused space will waste 1 MB of a disk capacity!) You need to find a balance between these two extremes.

### Field (Data) Type

You choose the type of data you'll want in each field, for example, text, number, date, currency, etc. This can prevent incorrect entries; for example, if you accidentally try to type someone's name into a numeric field, the computer may beep or flash an error message. **Trap!** Typing a number into a Text field is allowed, but the number becomes text which has a value of zero—you can't calculate with it! **Tip:** Use text-type fields to hold numbers that you never want to calculate with (zipcodes, phone numbers, etc.) so they are never accidentally calculated.

### Field Properties

You can add a variety of attributes to a field. For example, you could require a valid data range (say from 1 to 100) and any number entered outside of that range will be rejected. Or you could specify that phone numbers be entered in a certain way by creating an input mask, for example, (\_\_\_\_) \_\_\_\_-\_\_\_\_.

Here is the Farm File Structure Table:

Field Name	Field Type	Field Width
Plot #	Text	3
Worker	Text	10
Corn	Numeric	2
Beans	Numeric	2

**Trap!** The #1 trap for beginners is to enter *values* (for example, TOM instead of Worker) in the Field Name column.

## DATA (VALUES)

Data is words, numbers, symbols, graphics, video, and sound. You enter data into each field of the structure for each record. Here is the Farm File Data Table:

Plot #	Worker	Corn	Beans
1	TOM	4	6
2	SUE	3	7
3	JANE	10	2
4	PAUL	2	9
5	SAM	5	3
6	ANN	8	1

### Structure Table

Field *parts* across top.  
Each row defines one *field*.

Name Type Width

Field 1-----  
Field 2-----  
Field 3-----  
Field 4-----

### Data Table

Field *names* across top.  
Each row defines one *record*.

Field 1 Field 2 Field 3 Field 4

Record 1-----  
Record 2-----  
Record 3-----  
Record 4-----  
Record 5-----  
Record 6-----

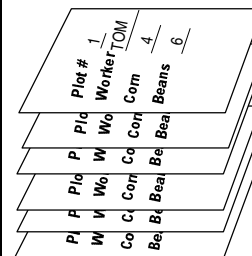
## TABLE (LIST, DATASHEET)

A table is a data entry grid of columns and rows.

A field name heads each column with field values listed in cells beneath. Each row represents a single record.

A table is the best layout for comparing, proofreading, and editing multiple records. However, unless you're careful, it's not best for data entry, because it's too easy to enter values into the wrong cell.

Plot #	Worker	Corn	Beans
1	TOM	4	6
2	SUE	3	7
3	JANE	10	2
4	PAUL	2	9
5	SAM	5	3
6	ANN	8	1



Answer these questions about the table shown above:

How many records are shown? \_\_\_\_\_

How many fields per record are shown? \_\_\_\_\_

What is the entity being described? \_\_\_\_\_

What is the name of Plot #1' s second field? \_\_\_\_\_

What is the name of Plot #2' s second field? \_\_\_\_\_

What is the name of the second field of ALL records? \_\_\_\_\_

What field part does each cell represent? \_\_\_\_\_

What value is stored in the second field of Plot #1 \_\_\_\_\_; Plot #2 \_\_\_\_\_; Plot #3 \_\_\_\_\_

Answers: 6; 4; Plot; Worker; Worker; Worker; Area; TOM; SUE; JANE.

## FORM

A form is a customized data entry screen.

Single-record columnar [CAWL-um-nar] (in other word, "in a column") forms let you focus on one record at a time. You can use the keyboard to [Page Up] or [Page Down] which fits the record-equals-page analogy. Multi-record tabular [TAB-uu-lur] (in other words, "table-like") forms display one record per row.

If desired, you can customize a form' s layout by changing field names and the order in which they appear to match a paper-based form. You can include data entry instructions next to a field (for example: The Plot # field is required), drop-down lists of acceptable field values, checkboxes, graphics, or whatever you desire to make data entry easier and more accurate.

<b>Plot Number</b>	1	
<b>Worker Firstname</b>	TOM	
<b>Acres of Corn</b>	4	
<b>Acres of Beans</b>	6	

Plot #	<u>1</u>
Worker	<u>TOM</u>
Corn	<u>4</u>
Beans	<u>6</u>

Page

## QUERY

A query (KWIH-ree / KWAIR-ree) is a set of instructions that search for and extract records from a database.

You can write these instructions directly in a query language, but most DBMSs let you fill in a grid with the field criteria you desire and then write the instructions for you behind the scenes. **Tip:** Query means to question.

Example: The grid shown here results in a table that displays two fields (Worker, Beans) and lists the records of farmers who've planted more than 5 acres of beans (TOM, SUE, PAUL).

**Data Mining** is a query technique that uses artificial intelligence (software) and neural networks (hardware) to discover unsuspected data relationships and detect underlying trends in a database.

Query		
Field Name	Worker	Beans
Show Field?	Yes	Yes
Criteria		>5

Worker	Beans
TOM	6
SUE	7
PAUL	9

**Query Results**

## REPORT

A report is a formatted display of database data.

You can create custom reports that automatically total, average, or otherwise manipulate field values. A report can generally contain any additional text you wish, including alternate field names for those used in the database.

Example: The depicted Farm Report automatically adds the total acreage of Corn and Beans planted.

### Labels

A label report prints database data on adhesive labels or directly to envelopes. For example, a mailing list of names and addresses could be printed on 3-across adhesive labels (depicted here) which can then be applied to envelopes.

### Farm Report

Number of Plots: 6

Corn Acres: 32

Bean Acres: 28


## RAM-BASED

A RAM-based DBMS copies *all* records from disk to RAM (all records must fit in RAM at once).

Pro: Because RAM is fast, database operations are fast.

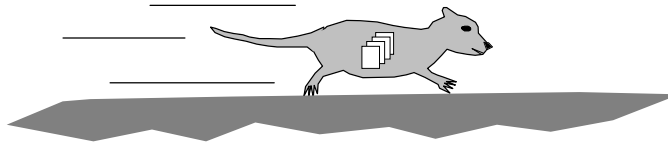
Con: Because RAM size is limited, database size is relatively tiny (several thousand records).

RAM-based databases are typically spreadsheets being used as databases where each row = 1 record.



RTF = a Rat is Tiny but Fast.

Observe the fast rat contains a tiny number of records.



## DISK-BASED

A Disk-based DBMS copies only some records from disk to RAM (swapping in other records as needed).

Pro: Because disk size can be large, database size is potentially large (several *billion* records).

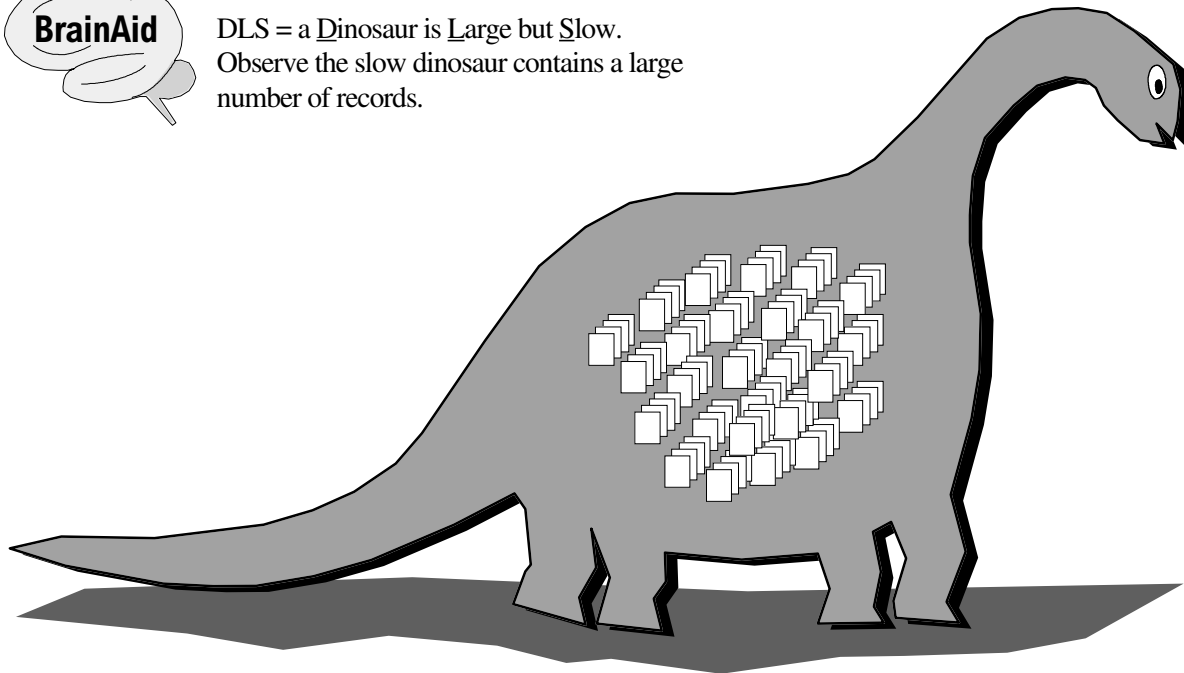
Con: Because swapping disk records is slow, database operations are relatively slow.

In general, standalone database applications are Disk-based.



DLS = a Dinosaur is Large but Slow.

Observe the slow dinosaur contains a large number of records.

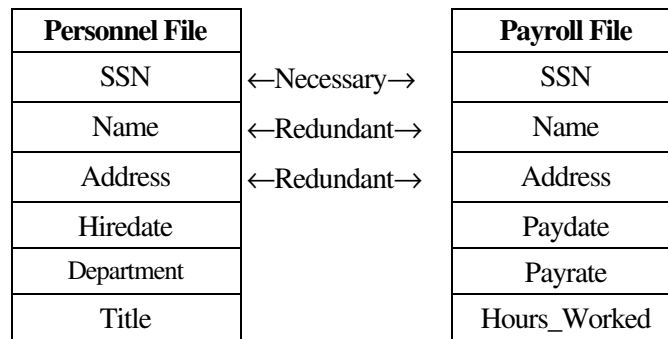


## FLAT-FILE (LIST MANAGER)

A flat-file DBMS can work with only one database file at a time.

Pro: Simple to learn and use for keeping track of things like book or video collections. Inexpensive.

Con: Flat files can NOT share data with each other. Files must contain redundant fields making update errors more likely.



### Redundancy Wastes Resources

Observe that three fields—SSN, Name, Address—are duplicated in the two flat files shown. The SSN (Social Security Number) is necessary in each file, because it uniquely identifies each employee. But the Name and Address fields are redundant. It wastes time, effort, and disk space to enter the same names and addresses more than once. And a firm typically has numerous employee files like: Performance File, Benefits File, Vacation File, Sick Leave File, etc.

### Redundancy Increases Errors

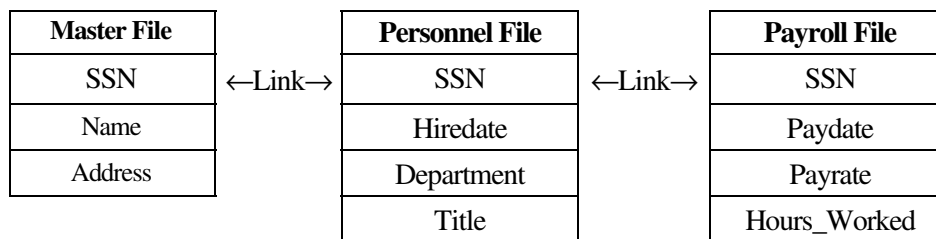
The more redundant data you have to update, the greater the chances of an error or omission. Let's say that you move to a new home and inform your Personnel Department. It updates your address but fails to inform the Payroll Department. Consequently, your paychecks get mailed to your old address. Or, suppose Personnel *does* inform Payroll but the data entry clerk transposes the numbers on your new address. Same problem.

## RELATIONAL\*

A relational DBMS can work with several files or tables at a time.

Pro: Relational files share data, reducing redundancy/errors. **BrainAid:** Relatives share family history.

Con: More complex, more expensive than a Flat file DBMS.



### Relational Reduces Redundancy

Observe that ONLY the SSN field is needed to electronically link the Master File to related Files. When the Personnel Department displays a record on screen, they see the three Master File fields followed by the three Personnel File fields. When the Payroll Department displays a record on screen, they see the three Master File fields followed by the three Payroll File fields.

### Relational Reduces Errors

Updating a Master File field updates all linked Files. Again let's say that you moved and informed the Personnel Department which enters your new address into the Master File. This entry then automatically flows to both the Personnel and the Payroll Files. Of course, one Master File data entry error affects *every* linked file, but it's just as easily fixed.

**Note:** Instead of linking separate files, some DBMS' s link tables within a single file; e.g., Master Table, Personnel Table, Payroll Table.

## DB SUMMARY



Look back through these pages and write down a brief description that will help you remember each term.

File—

Record—

Field—

Key\*—

DBMS—

Structure (Design)—

Data (Values)—



Table

Form

Query

Report

RAM-Based—

Disk-Based—

Flat-File (List Manager)—

Relational\*—

# DB TermMap



Practice reciting the following terms in order from memory. Then briefly define each.  
**Terms (14):** File, Record, Field, Key, DBMS, Structure, Table, Form, Query, Report, RAM-Based, Disk-Based, Flat-File, Relational.

**Acronym:** FRF-KD-STF-QR-RD-FR

**Acrostic:** Furf kid stuf qur red fur.

Plot #	<u>1</u>
Worker	<u>TOM</u>
Corn	<u>4</u>
Beans	<u>6</u>

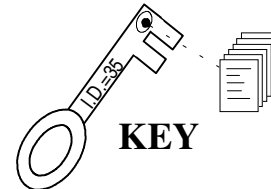
**FILE**

Plot #	<u>1</u>
Worker	<u>TOM</u>
Corn	<u>4</u>
Beans	<u>6</u>

**RECORD**

Plot # 1

**FIELD**



**KEY**



**DBMS**

Field	Field Name	Field Type	Field Width
1	Plot #	Text	2
2	Worker	Text	10
3	Corn	Numeric	4
4	Beans	Numeric	4

**STRUCTURE**

Plot #	Worker	Corn	Beans
1	TOM	4	6
2	SUE	7	5
3	JANE	10	2
4	PAUL	2	9
5	SAM	5	3
6	ANN	8	1

**TABLE**

Plot #	1
Worker	TOM
Corn	4
Beans	6

**FORM**

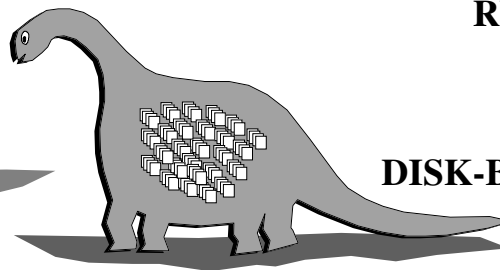
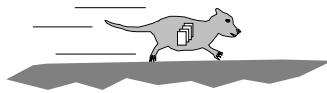
Query		
Field Name	Worker	Beans
Show Field?	Yes	Yes
Criteria		>5

**QUERY**

Farm Report	
Number of Plots: 6	
Corn Acres: 32	
Bean Acres: 28	

**REPORT**

**RAM-BASED**



**DISK-BASED**

Personnel File	
SSN	
Name	
Address	
Hiredate	
Department	
Title	

←Necessary→  
 ←Redundant→  
 ←Redundant→

Payroll File	
SSN	
Name	
Address	
Paydate	
Payrate	
Hours_Worked	

Master File	
SSN	
Name	
Address	

←Link→

Personnel File	
SSN	
Hiredate	
Department	
Title	

←Link→

Payroll File	
SSN	
Paydate	
Payrate	
Hours_Worked	

**FLAT-FILE**

**RELATIONAL**