

Contents

[Introduction](#)

[Functions](#)

[Commands](#)

[Support](#)

Introduction

The NoteBook Program Copyright © 1994 by H. D. Baecker

NoteBook is a program designed to store, edit, and retrieve information from notes made in the course of fieldwork under various conditions by users in a variety of occupations. The program was conceived as an aid to persons without any computer expertise or experience, it is as appropriate for keeping case notes in the course of social work as it for use on a portable computer to record field notes in the during an archeological dig. In fact its uses range from recording an everyday diary to keeping and using records of professional consultations.

The unit of entry, editing, etc., is a Note, such as this Introduction. Every Note is dated by the program. The user is invited to enter a title, of up to 40 characters, for each Note. The user is then free to enter the text of the Note, which may be up to some 32,000 characters long. That limit is about 80 pages of a small-print paperback book.

Notes are added to Notebooks, which correspond to DOS files, and which may be searched to locate those Notes that contain references specified by the user at the time of the search and need never be known in advance when the Note is created. A Notebook may hold 16,380 Notes, but that is a theoretical limit, think of about 1,000 Notes as being the practical limit.

Indexing and related topics and user instructions are covered in other Help topics, see Functions and Commands.

Functions

[Indexing](#)

[Unwords](#)

[Conflation](#)

[Queries](#)

Indexing

Every Note is indexed by the program when the Note is added or edited. Every word used in the Note is indexed, unless that word is an "unword". Unwords are those words of the language that it makes no sense to use as index terms, like "the" and "and". Please see Unwords for a detailed explanation.

The remaining words are conflated, that is, they are reduced to their word-stem, wherever this action is applicable. For instance, regular plurals in "-s" are reduced to their singular forms, the past tense of verbs in "-ed" is changed to the present tense, and so on. One copy of each stem, however many times it occurs in the text of the Note, is then stored in an index associated with that Note. For further discussion please see Conflation.

When a user queries a Notebook the index of each of its Notes is examined to seek a match for the query term, and the Notes satisfying the specified criteria are marked. These may be individually displayed for inspection by using the Notes|Show command. The procedure is explained in detail under Queries.

Unwords

In any language there are a large number of words that do not qualify as index terms for information retrieval purposes because they are used so frequently that they can be found in just about any text of the language. There are the immediately obvious words such as "the" and "and" , but there are quite a few more, such as "but" and "about" and "this" and "such" itself, etc. There is as yet no generally accepted name for these words, so we call them Unwords.

The NoteBook program comes with a distribution file of about 200 of these unwords. This file is loaded every time the program is executed. When the user completes the addition of a Note by saving it (using the command Notes|Save, or by selecting another action) then the Note is indexed, but first the unwords in the Note are eliminated from consideration for indexing (but they remain present in the stored text as entered). The remaining words are then conflated and a single copy of each root or word-stem is stored in the index for that Note. When a Notebook is queried the search is of each Note's index to locate and mark those Notes that meet the criteria.

Users are free to edit the file of unwords, UNWORDS.DTA, using an ASCII-only text editor such as MS-Windows Notepad or MS-DOS Edit. Often a user may wish to add words, usually dialect or slang, that are locally insignificant, equally as often a user may wish to delete a certain word so that it is not considered by the program to be an unword. For instance, the distribution file contains "so" as an unword. But a social worker with a client base of a large number of ethnic Chinese may have to delete "so" as an unword because "So" may be a widespread family name and thus essential as an index term.

The distribution file is in alphabetical order, this is only for convenience for editing and is not strictly necessary, neither is it necessary that entries be only in upper case. But there is a requirement that entries contain no punctuation, that is, the correct unword rendering of "don't" is "dont" or "DONT" in the file. There is no restriction on the length of unwords, nor is there a strict limit to their number. The file of unwords in current use must always have the filename UNWORDS.DTA and must always reside in the same subdirectory as the NOTEBOOK.EXE file.

Increasing the number of unwords increases the time it takes to determine for each word in a Note whether it is an unword or not, but decreases the number of words to be conflated and indexed subsequently, and so decreases the time to search an index during retrieval. Increasing the number of unwords stored in the program also subtracts from the space available for other purposes during program use. You choose.

Conflation

Conflation is the process, in linguistics, of reducing a word to its stem or root, often referred to as "stemming". This process can be carried out for agglutinative languages, such as English, much more so Finnish, that accrete tense or case designations to word roots. Analytic languages, such as those of South-East Asia, express shades of meaning by using strings of separate words.

Conflation would work perfectly if we used a language that admitted no irregular words or any exceptions. But in English those who have more than one child have children, not childs, and if we go to the park today we went there yesterday perhaps, or might have gone. What we did not do is goed there.

Reality is. So conflation is an imperfect tool that works for the regular nouns and verbs of English, the majority, and does not work for the irregular ones.

Most queries will be based on nouns or noun and adjective combinations. It may seem that the way verbs are handled does not matter much. This is true to some extent, but, particularly in English, it is not possible by simple mechanical text analysis to establish whether a word is or is not a verb. The program might as well attempt to handle verbs efficiently too to conserve space and time, mixed in with the rest.

But in framing queries the user is advised to take account of irregular verbs and nouns, and of the misleading trail they might leave in the index.

Queries

Searches of the current Notebook are activated using the Queries|Query menu command, which brings up the queries dialog.

Query terms entered across a line of the dialog are used in conjunction for the search, that is, if you enter "bright objects" on the same line of the dialog in consecutive entry fields (no gaps allowed) then for a Note to be retrieved it must contain both words, but these words do not need to occur together in the Note text or title. Thus "bright objects" will find the occurrence of both or either of the sequences "bright objects" and "objects that are pink and bright" in a Note, or any Note that contains both words at least once each anywhere in the Note. Or, of course, allowing for conflation, the words "object" or "objected" and "bright".

Query terms or groups of terms on separate lines of the query dialog form independent queries. Up to 5 query lines at a time may be entered, each of up to 4 terms. Each term must be a single word, no embedded spaces allowed, so that if you are looking for "John Hancock" you must enter "John" and "Hancock" separately on the same line. If you need to find references to both "John Hancock" and "Edwin Smith", together in the same Note, enter all 4 words on a line. However, if you want to find references to either of them, enter "John" and "Hancock" on one line and "Edwin" and "Smith" on another.

To find Notes referring to both "John Hancock" and "John Jones" you need only enter the three terms "John", "Hancock" and "Jones" on a line.

Always enter terms from the left margin on a line, and from the top down, leaving no gaps, so that unused entry fields are at bottom right of the dialog.

The start and end dates for the search must be entered as numbers only (e.g. March is "3") in the order year (all 4 digits, life is confusing otherwise around the turn of the century), month, day.

Before the search each query terms is conflated, but the program does not check for unwords in queries.

After a Query the list of "hits" is displayed and any Note on it can be selected for display. Later use Notes|Show to display other Notes found by the search. Found Notes are so displayed until cancelled by using Query|Cancel. Subsequent Note lists will again be of Active Notes.

Commands

[File Menu](#)

[Notes Menu](#)

[Queries Menu](#)

[Edit Menu](#)

File Menu

New

Sets up a new Notebook. As Notebooks are embodied as MS-DOS files Notebook names follow the limitations of DOS filenames, but are entered without extensions. Immediately after creating a new Notebook the first use of it must be to add at least one Note to it.

Open

Opens an existing Notebook for use. Call File|Open and then click Cancel in the Files Dialog box if you wish to see the display of the Current Directory name, or if you wish just to browse the available Notebooks in the Current or other directories.

Directory|Change

Changes to the nominated subdirectory. Must be used before File|New to create a new Notebook in an existing subdirectory other than the Current Directory.

Directory|Make

Makes a new subdirectory and then changes to it as the Current Directory.

Compact

Eliminates the gaps caused by editing Notes and physically removes previously hidden Notes from the current Notebook, returning a compacted Notebook file. If the user does not name an archive file then no Notes are deleted and hidden Notes are instead returned to view., and only gaps resulting from editing are recovered.

Exit

Exits from the program, first storing the current Note if in the course of being Added or Edited, and closing the open Notebook if there is one. Note that every Add or Edit causes the Index on disk to be updated immediately.

Notes Menu

Add

Displays an input dialog for the user to enter a title of up to 40 characters. The program will continue to cycle until a title has been entered. Then it enables entry of the body of the Note. A Note may comprise up to 32,766 characters, including spaces and all other punctuation. The Enter key causes two characters to be entered, but users are advised to use it only at the end of paragraphs and to rely on the wordwrap feature, always enabled, otherwise the text will not accommodate to different window sizes. Entry of the Note is terminated by choosing Notes|Save or by choosing another activity.

Show

Displays a list of the Notes in the current Notebook. Select a Note to see by clicking on it and OK, or double-clicking on the Note. After viewing the displayed Note click on Edit|Clear or start another action to clear the screen of it. The Note list displayed is normally of all the active Notes. But after a query, until Query|Cancel has been selected, the display is of those Notes that satisfied the most recent query. When Notes|Restore is selected then the display is the list of Hidden Notes.

Edit

Choose the Note to edit from the list. After editing use Notes|Save, or start another activity, to store the edited Note, or choose Edit|Clear to retain the original unedited version of the Note. Whatever the changes you appear to make on the screen you cannot change the original stored date or title.

Save

Indexes and saves the current Note to the open Notebook and clears the screen. It has no effect unless used immediately after an Add, Edit, or Input action.

Hide

Hides the selected Note. A hidden Note is not listed except to Restore it, does not get searched during a query, and becomes a candidate for archiving and deletion when the Notebook is Compacted.

Restore

Select and Restore a Hidden Note to view. The Restored Note again becomes a candidate for queries.

Input

Reads a Note from an external file specified by the user. Notes read are assumed to have the input format explained under Input from File.

Print

Prints the currently displayed Note on the default printer. There is an attempt at pagination but no other layout niceness. To format the text for prettyprint either SaveAs the Note to a file, or CopyAll to the Clipboard, and then massage it with a text editor or word processor of choice.

SaveAs

Outputs the Note as ASCII text to the chosen file. If the file already exists the Note is appended, otherwise a new file is created.

Queries Menu

Query

Initiates a search of the current Notebook as explained under Queries.

Cancel

Cancels the results of the most recent search so that "hits" are forgotten. It is not necessary to Cancel before initiating another search with Queries|Query but until a search result is cancelled ordinary display and editing of Notes is prevented as all Active Notes are not shown in the Notes list.

Edit Menu

Cut

Copies the selected text to the Clipboard and cuts, clears, that text from the screen.

Copy

Copies the selected text from the screen to the Clipboard.

Paste

Copies the contents of the Clipboard to the screen at the cursor.

Clear

Clears the screen and terminates processing of the current Note.

CopyAll

Copies all of the current Note to the Clipboard.

Support

Via BBS

Call the BMT Micro BBS at 910.350.8061

At the Main Menu choose Shareware, then Notebook and leave your message and/or read the replies.

By Mail

Write to:

H D Baecker

Box 474

Lake Cowichan BC V0R 2G0

Canada

Input from File

The menu commands File|Compact (if deleted Notes are archived) and Notes|SaveAs output Notes in a form that may be re-entered using Notes|Input. Such Notes may also be generated offline using a text editor such as MS-DOS Edit or MS_Windows Notepad, that is, one that inserts no formatting characters other than the ones the user types at the keyboard.

The format for a Note to be input from a file is as follows:

Zero, one, or more lines of text that are ignored by the input routine. Then a line the first 8 characters of which are "Title:- ", that is, upper case T, the lower case characters "itle", and then ":-" followed by a space. The remaining text on that line is taken to be the title of the Note, up to 40 characters long. The lines following are considered as the text body of the Note.

If the input file holds only a single Note the simple sequence Notes|Input, Notes|Save, will Add the Note to the current Notebook, with the current date.

If an input file holds more than one Note then Notes|Input will read the whole file into the window, if it is within the size limit. The user must then edit the text with the Edit menu commands using the Clipboard as temporary storage to bring each Note into the window separately for Notes|Save to store it.

No Note may exceed 32767 characters in length. No input file of more than one Note may exceed that length either. If a longer file is to be read then it must first be split up using a text editor program.

Spelling!

There is English spelling. There is American spelling. There is no doubt other spelling. And then there is spelling. In some way or other any information retrieval facility using text depends on spelling.

The NoteBook program only depends on "correct" spelling to eliminate unwords from indexing, and on the spelling of word endings to implement conflation.

Insofar as possible the conflation process caters to both English and American spelling conventions, but if the English conventions fail then please revert to American usage.

However, the program is utterly dependent on CONSISTENT spelling by the user in order to make effective retrieval possible.

The program will "work" after a fashion for any language that standard Windows can handle as input. The ability to eliminate unwords is not essential to the operation of the program and can be retained by generating a foreign UNWORDS.DTA file. The English conflation rules (not amenable to user editing) are not essential either but may generate some strange results in some languages. In particular, languages that have accented characters or rely upon punctuation marks to augment spelling may store some bizarre text and might generate some even more bizarre index entries.

Commands

conflated

Conflation

Conflation.

Functions

Indexing

Notes

queries

Unwords

