

The SET COLOR TO command

After you load into memory your custom colors, you tell dBASE to use those colors with a single command. Just enter

```
SET COLOR TO &xMyColors
```

When you do, dBASE uses the contents of the variable xMyColors to determine which colors to use for your display.

Applying the technique

Suppose you define a number of color schemes and save them under the names COLOR1, COLOR2, and COLOR3. You can save all of the color schemes in a single file by entering a command in the form

```
SAVE TO MYCOLORS ALL LIKE COLOR?
```

Then, when you enter the RESTORE FROM command to load into memory the variables you saved in MYCOLORS, you can use the SET COLOR TO command to activate any of the three color schemes on an as-needed basis.

Conclusion

dBASE makes it easy to customize your screen colors for a given work session. However, if you want to change the screen colors on a permanent basis, you must edit your CONFIG.DB file. In this article, we showed you how to save custom color settings in memory variables. By saving the color settings in a disk file, you can activate a custom color scheme with a single SET COLOR TO command. ♦

DEVELOPER'S CORNER



Using the plus and minus keys to change dates

By Vincent D. O'Connor

Many commercial applications allow you to increment or decrement a date field by pressing the plus and minus keys. That feature is popular because it saves time and makes the program easier to use. Although dBASE doesn't provide a built-in way to change dates by pressing the plus or minus key, you can add that feature to your applications with the routines we present in this article. You can use this technique whether you use dBASE IV or dBASE 5.

The DATECHNG program

In order to demonstrate the procedures and functions that allow you to use the plus and minus keys to edit a date field, create and run the DATECHNG program shown in Listing A on the next page. Note that the DATECHNG program includes listings for two functions and two procedures.

When you run DATECHNG, it displays the prompt *Enter date:* and the current system date, as shown in Figure A. Press plus (+) or minus (-) to add or subtract a day from the date. You can use the keys on the top row or on the numeric keypad.

By the way, you can type a date directly into the entry field, if you prefer. However, when you change

a date by pressing the plus or minus key, you'll notice that the procedures change the color of the date field to black on white. You exit the program by pressing [Enter].

Figure A

```
Enter date: 16/07/95
```

```
DATECHNG
```

The DATECHNG program lets you change a date by pressing the plus and minus keys.

How DATECHNG works

The essence of our technique is, of course, trapping the plus and minus keys when the cursor is in the date field. However, in your application, you'll want the plus and minus keys to work normally when the cursor is in a character or numeric field.

The trick is to enable the special handling of the plus and minus keys just prior to entering a date field and disable that special handling once the cursor leaves the field. You activate the special handling by adding WHEN and VALID REQUIRED clauses to the @...GET commands that refer to your date fields.

Using WHEN to enter the date field

The WHEN clause allows you to ensure that certain conditions are met before the cursor enters the GET field. Typically, you use the WHEN clause to keep the cursor out of certain fields, depending on the values the operator enters in another field.

The WHEN clause will accept as a condition a user-defined function (UDF) as long as the function returns a logical value—true (.T.) or false (.F.). Therefore, you can associate with your WHEN clause a UDF to do clean-up processing before the cursor enters the field. In DATECHNG, the WHEN clause calls a function that allows special handling of the plus and minus keys so that the operator can use those keys to change the entry in a date field.

Using VALID on leaving the date field

The VALID clause ensures that certain conditions have been met before the cursor leaves the GET field. Normally, you use a VALID clause to identify and keep out errors in data entry. VALID keeps the cursor in the GET field until the entry meets a certain condition.

Like WHEN, VALID accepts as a condition a UDF that returns a logical value. So, you can use the VALID clause to do any work you need done before the cursor leaves the date field. In DATECHNG, the VALID clause calls a function to cancel the special handling of the plus and minus keys.

The REQUIRED option

The REQUIRED option of the VALID clause forces dBASE to call the DATE_OFF() UDF, which cancels the special handling of the plus and minus keys. Without the REQUIRED option, the user could press [Esc] to exit the GET field and bypass the VALID check. When you include the REQUIRED option, even if the user presses [Esc], the entry field respects the VALID clause.

The DATE_ON() and DATE_OFF() UDFs

When you call the DATE_ON() function, it associates the procedure DATEPLUS with the plus key and the procedure DATEMINUS with the minus key. The key to this technique is customizing your dBASE keyboard with the ON KEY LABEL command, which allows you to trap certain keys and ensure that dBASE will execute a particular procedure when the operator presses a particular key. After DATE_ON() assigns the procedures to the plus and minus keys, it returns a logical true.

The DATE_OFF() function simply turns off the special handling of the plus and minus keys. When you issue the ON KEY LABEL command without specify-

Listing A

- DATECHNG.PRG by Vincent D. O'Connor
- VERSION: dBASE IV, dBASE 5
- PURPOSE: Demonstrate the use of the plus
• minus keys to increment and decrement
• the entry in a date field.

```
@ 0,0 CLEAR
CLEAR ALL
SET TALK OFF
SET BELL ON
SET EXACT ON
SET CONFIRM ON
SET SCOREBOARD OFF
SET ESCAPE OFF
date = DATE()
disprow = 12
dispcol = 12
@ 12,0 SAY "Enter date:" GET mdate ;
  WHEN DATE_ON() ;
  VALID REQUIRED DATE_OFF()
READ
CLEAR GETS
@ 14,0 SAY mdate
RETURN
```

```
FUNCTION date_on
ON KEY LABEL + DO DATEPLUS
ON KEY LABEL - DO DATEMINUS
RETURN .T.
```

```
FUNCTION date_off
ON KEY LABEL +
ON KEY LABEL -
RETURN .T.
```

```
PROCEDURE dateplus
PRIVATE get_var
get_var = VARREAD()
&get_var = &get_var + 1
@ disprow, dispcol SAY &get_var COLOR N/W,
RETURN
```

```
PROCEDURE dateminus
PRIVATE get_var
get_var = VARREAD()
&get_var = &get_var - 1
@ disprow, dispcol SAY &get_var COLOR N/W,
RETURN
```


ing a procedure, dBASE restores the key to its default status. That is, when you press the plus and minus keys, they'll no longer call any special procedures.

The DATEPLUS and DATEMINUS procedures

The DATEPLUS and DATEMINUS procedures do the actual work of incrementing and decrementing the date value. In each procedure, the VARREAD() function returns the name of the variable. Then, the procedures simply add or subtract 1 from the value of the named variable. Finally, the procedures display the new date value.

The procedures use two public variables, *disprow* and *dispcol*, to determine at what row and column the changed date should appear. You use this approach because dBASE doesn't automatically reflect the changed value of the variable in the GET field. In your application, remember to set the value of *disprow* and *dispcol* to the row and column where the date variable will appear prior to calling DATE_ON() and DATE_OFF() in your GET command.

Incorporating the plus and minus date change feature into your application

The DATECHNG program includes two UDFs and two procedures that work together to allow date changes with the plus and minus keys. By including the code for the UDFs and procedures in your active procedure file, you can offer this feature in any of your date fields with a command in the form

```
•...SAY...GET dateval WHEN DATE_ON() VALID REQUIRED  
DATE_OFF()
```

where *dateval* is the name of your database field or memory variable.

A date of a different color

As we mentioned, when you change a date by pressing the plus or minus key, the DATEPLUS and DATEMINUS procedures change the color of the date field to black on white. You may want to display the date in one color if the operator presses the plus key and another color if the operator presses the minus key. To do so, simply edit one of the procedures and change the command

```
• disprow, dispcol SAY &get_var COLOR N/W,
```

replacing N/W, with the color code of your choice.

Adding and subtracting more than a day at a time

In some cases, you may want to add or subtract more than a single day to the date. For instance, if you're writing a scheduling application, you may want to add or subtract a week from a given date when the operator presses plus or minus. To do so, you'd modify the procedure DATEPLUS and edit the line

```
&get_var = &get_var + 1
```

by changing the 1 to 7. Of course, you'd also edit the DATEMINUS procedure and change the line that subtracts 1 from the variable.

Conclusion

Many commercial applications spoil end users by letting them press the plus and minus keys to change the value in a date field. In this article, we showed you one way to implement this feature in your dBASE IV and dBASE 5 applications. In a future issue, we'll show you how to apply this technique to increment or decrement a numeric field with a single keystroke. ♦

Vincent D. O'Connor is a dBASE developer from Babbitt, Minnesota. You can reach him at (218) 827-2787.

Changing dates with the up and down arrow keys

In the accompanying article, we show you how to allow an operator to change a date by pressing the plus or minus key. The key to our technique is using the ON KEY LABEL command to tell dBASE to increment or decrement a date when the operator presses a particular key.

You can apply the same technique to allow an operator to change a date by pressing either ↑ or ↓. To do so, simply edit the user-defined function DATE_ON() and change the command

```
ON KEY LABEL + DO DATEPLUS
```

to

```
ON KEY LABEL UPARROW DO DATEPLUS
```

In like manner, to decrement a date when the operator presses ↓, change the command

```
ON KEY LABEL - DO DATEMINUS
```

to

```
ON KEY LABEL DNARROW DO DATEMINUS
```

In addition, you must edit the commands in DATE_OFF(), replacing the plus (+) and minus (-) signs with UPARROW and DNARROW, respectively.