# HOW TO STORE VARIABLE INFORMATION ON DISK

When I write a program, I sometimes find it easier to store variables on disk rather than inside the program. Using a few variables, it's okay to store them in a program but what about storing lots of information in variables. This makes your program a lot longer. Let's say, for example, I was writing an adventure game and I wanted 100 location descriptions. In my game, I could have a routine like this.

```
10 dim LOCATION$(100) 20 for X=1 to 100 : read LOCATION$(x) : next X

1000 rem LOCATION DESCRIPTIONS

1010 data "This is location one."

1020 data "This is location two."
```

And so on, adding a hundred lines to STOS. How about doing it this way?

```
10 LOCATION=1

20 print "Type in the description for location "+str$(LOCATION)

30 input L$

40 F$=file select$("*.DAT")

50 open out #1,F$

60 print #1,L$

70 close #1

80 cls : inc LOCATION : goto 20
```

Doing things this way with this separate routine allows me to enter the description as I would see it on screen, all nicely spaced out, rather than playing about spacing out the words in a data statement. I could save each file out under names like LOC1.DAT for location one and LOC2.DAT for location two etc. Once all descriptions are saved then I could put a routine in the game to load each file when I need it. For example.

```
10 LOCATION=1

15 F$="LOC"+str$(LOCATION)-" "+".DAT"

20 open in #1,F$
```

30 input #1,L$

40 close #1

50 print L$ : rem print Location

60 print : inc LOCATION : goto 15

Here's another tip for a program that needs loads of zones on-screen on loads of pictures. Just simply type out a program that allows you to create the zone data using just the mouse.

10 key off : flash off : curs off : mode 0

15 dim X1(10,10),Y1(10,10),X2(10,10),Y2(10,10)

20 ZP=1 : ZA=1 : rem ZP=Zone Picture ZA=Zone Area

30 F$="PIC"+str$(ZP)-" "+".PI1?

40 load F$ : screen copy physic to back

50 repeat : until mouse key=1 : X1=xmouse : Y1=ymouse

60 wait 40

70 repeat : until mouse key=1 : X2=xmouse : Y2=ymouse

80 rem X1,Y1 : Top left hand co-ordinates of zone

90 rem X2,Y1 : Bottom right hand co-ordinates of zone

100 X1(ZP,ZA)=X1 : Y1(ZP,ZA)=Y1 : X2(ZP,ZA)=X2 : Y2(ZP,ZA)=Y2

110 if ZA<10 then inc ZA : goto 50

120 if ZA=10 and ZP<10 then inc ZP : ZA=1 : goto 50

130 open out #1,"ZONE.DAT"

140 for X=1 to 10 : for Y=1 to 10

150 print #1,X1(X,Y),Y1(X,Y),X2(X,Y),Y2(X,Y)

160 close #1

Once this is done you simply load this file into your program using 'Open In' and you can set your zones easier. For example.

10 key off : curs off : flash off : mode 0

```
20 dim X1(10,10),Y1(10,10),X2(10,10),Y2(10,10)

30 open in #1,"ZONE.DAT"

40 for X=1 to 10 : for Y=1 to 10

50 input #1,X1(X,Y),Y1(X,Y),X2(X,Y),Y2(X,Y)

60 close #1

70 ZP=1

80 unpack 5 : rem unpack picture one

90 for ZA=1 to 10

100 set zone ZA,X1(ZP,ZA),Y1(ZP,ZA) TO X2(ZP,ZA),Y2(ZP,ZA)

110 next ZA

120 repeat : CH=zone(0) : until mouse key=1
```

There are loads of things you can do to improve the length of your programs and makes things easier. One thing you can do is write a routine to record the X and Y coordinates of alien movement instead of typing in the coordinates in data statements by hand.